Peter Baumann, Jacobs University Bremen / Rasdaman GmbH Bremen, Germany
Gabriele Bavota, University of Salerno, Italy
Grigorios N. Beligiannis, University of Western Greece, Greece
Noureddine Belkhatir, University of Grenoble, France
Jorge Bernardino, ISEC - Institute Polytechnic of Coimbra, Portugal
Rudolf Berrendorf, Bonn-Rhein-Sieg University of Applied Sciences - Sankt Augustin, Germany
Ateet Bhalla, Independent Consultant, India
Fernando Boronat Seguí, Universidad Politecnica de Valencia, Spain
Pierre Borne, Ecole Centrale de Lille, France
Farid Bourennani, University of Ontario Institute of Technology (UOIT), Canada
Narhimene Boustia, Saad Dahlab University - Blida, Algeria
Hongyu Pei Breivold, ABB Corporate Research, Sweden
Carsten Brockmann, Universität Potsdam, Germany
Antonio Bucchiarone, Fondazione Bruno Kessler, Italy
Georg Buchgeher, Software Competence Center Hagenberg GmbH, Austria
Dumitru Burdescu, University of Craiova, Romania
Martine Cadot, University of Nancy / LORIA, France
Isabel Candal-Vicente, Universidad del Este, Puerto Rico
Juan-Vicente Capella-Hernández, Universitat Politècnica de València, Spain
Jose Carlos Metrolho, Polytechnic Institute of Castelo Branco, Portugal
Alain Casali, Aix-Marseille University, France
Yaser Chaaban, Leibniz University of Hanover, Germany
Savvas A. Chatzichristofis, Democritus University of Thrace, Greece
Antonin Chazalet, Orange, France
Jiann-Liang Chen, National Dong Hwa University, China
Shiping Chen, CSIRO ICT Centre, Australia
Wen-Shiung Chen, National Chi Nan University, Taiwan
Zhe Chen, College of Computer Science and Technology, Nanjing University of Aeronautics and Astronautics, China PR
Po-Hsun Cheng, National Kaohsiung Normal University, Taiwan
Yoonsik Cheon, The University of Texas at El Paso, USA
Lau Cheuk Lung, INE/UFSC, Brazil
Robert Chew, Lien Centre for Social Innovation, Singapore
Andrew Connor, Auckland University of Technology, New Zealand
Rebeca Cortázar, University of Deusto, Spain
Noël Crespi, Institut Telecom, Telecom SudParis, France
Carlos E. Cuesta, Rey Juan Carlos University, Spain
Duilio Curcio, University of Calabria, Italy
Mirela Danubianu, "Stefan cel Mare" University of Suceava, Romania
Paulo Asterio de Castro Guerra, Tapijara Programação de Sistemas Ltda. - Lambari, Brazil
Cláudio de Souza Baptista, University of Campina Grande, Brazil
Maria del Pilar Angeles, Universidad Nacional Autonónoma de México, México
Rafael del Vado Vírseda, Universidad Complutense de Madrid, Spain
Giovanni Denaro, University of Milano-Bicocca, Italy
Nirmit Desai, IBM Research, India
Vincenzo Deufemia, Università di Salerno, Italy
Leandro Dias da Silva, Universidade Federal de Alagoas, Brazil
Javier Diaz, Rutgers University, USA
Nicholas John Dingle, University of Manchester, UK
Roland Dodd, CQUniversity, Australia
Aijuan Dong, Hood College, USA
Suzana Dragicevic, Simon Fraser University- Burnaby, Canada
Cédric du Mouza, CNAM, France

Dayang Norhayati Abang Jawawi, Universiti Teknologi Malaysia (UTM), Malaysia
Jinyuan Jia, Tongji University. Shanghai, China
Maria Joao Ferreira, Universidade Portucalense, Portugal
Ahmed Kamel, Concordia College, Moorhead, Minnesota, USA
Teemu Kanstrén, VTT Technical Research Centre of Finland, Finland
Nittaya Kerdprasop, Suranaree University of Technology, Thailand
Ayad ali Keshlaf, Newcastle University, UK
Nhien An Le Khac, University College Dublin, Ireland
Sadegh Kharazmi, RMIT University - Melbourne, Australia
Kyoung-Sook Kim, National Institute of Information and Communications Technology, Japan
Youngjae Kim, Oak Ridge National Laboratory, USA
Cornel Klein, Siemens AG, Germany
Alexander Knapp, University of Augsburg, Germany
Radek Koci, Brno University of Technology, Czech Republic
Christian Kop, University of Klagenfurt, Austria
Michal Krátký, VŠB - Technical University of Ostrava, Czech Republic
Narayanan Kulathuramaiyer, Universiti Malaysia Sarawak, Malaysia
Satoshi Kurihara, Osaka University, Japan
Eugenijus Kurilovas, Vilnius University, Lithuania
Alla Lake, Linfo Systems, LLC, USA
Fritz Laux, Reutlingen University, Germany
Luigi Lavazza, Università dell'Insubria, Italy
Fábio Luiz Leite Júnior, Universidade Estadual da Paraiba,Brazil
Alain Lelu, University of Franche-Comté / LORIA, France
Cynthia Y. Lester, Georgia Perimeter College, USA
Clement Leung, Hong Kong Baptist University, Hong Kong
Weidong Li, University of Connecticut, USA
Corrado Loglisci, University of Bari, Italy
Francesco Longo, University of Calabria, Italy
Sérgio F. Lopes, University of Minho, Portugal
Pericles Loucopoulos, Loughborough University, UK
Alen Lovrencic, University of Zagreb, Croatia
Qifeng Lu, MacroSys, LLC, USA
Xun Luo, Qualcomm Inc., USA
Stephane Maag, Telecom SudParis, France
Ricardo J. Machado, University of Minho, Portugal
Maryam Tayefeh Mahmoudi, Research Institute for ICT, Iran
Nicos Malevris, Athens University of Economics and Business, Greece
Herwig Mannaert, University of Antwerp, Belgium
José Manuel Molina López, Universidad Carlos III de Madrid, Spain
Francesco Marcelloni, University of Pisa, Italy
Eda Marchetti, Consiglio Nazionale delle Ricerche (CNR), Italy
Gerasimos Marketos, University of Piraeus, Greece
Abel Marrero, Bombardier Transportation, Germany
Adriana Martin, Universidad Nacional de la Patagonia Austral / Universidad Nacional del Comahue, Argentina
Goran Martinovic, J.J. Strossmayer University of Osijek, Croatia
Paulo Martins, University of Trás-os-Montes e Alto Douro (UTAD), Portugal
Stephan Mäs, Technical University of Dresden, Germany
Constandinos Mavromoustakis, University of Nicosia, Cyprus
Jose Merseguer, Universidad de Zaragoza, Spain
Seyedeh Leili Mirtaheri, Iran University of Science & Technology, Iran
Lars Moench, University of Hagen, Germany
Yasuhiko Morimoto, Hiroshima University, Japan

Antonio Navarro Martín, Universidad Complutense de Madrid, Spain
Filippo Neri, University of Naples, Italy
Muaz A. Niazi, Bahria University, Islamabad, Pakistan
Natalja Nikitina, KTH Royal Institute of Technology, Sweden
Roy Oberhauser, Aalen University, Germany
Pablo Oliveira Antonino, Fraunhofer IESE, Germany
Rocco Oliveto, University of Molise, Italy
Sascha Opletal, Universität Stuttgart, Germany
Flavio Oquendo, European University of Brittany/IRISA-UBS, France
Claus Pahl, Dublin City University, Ireland
Marcos Palacios, University of Oviedo, Spain
Constantin Paleologu, University Politehnica of Bucharest, Romania
Kai Pan, UNC Charlotte, USA
Yiannis Papadopoulos, University of Hull, UK
Andreas Papasalouros, University of the Aegean, Greece
Rodrigo Paredes, Universidad de Talca, Chile
Päivi Parviainen, VTT Technical Research Centre, Finland
João Pascoal Faria, Faculty of Engineering of University of Porto / INESC TEC, Portugal
Fabrizio Pastore, University of Milano - Bicocca, Italy
Kunal Patel, Ingenuity Systems, USA
Óscar Pereira, Instituto de Telecomunicacoes - University of Aveiro, Portugal
Willy Picard, Poznań University of Economics, Poland
Jose R. Pires Manso, University of Beira Interior, Portugal
Sören Pirk, Universität Konstanz, Germany
Meikel Poess, Oracle Corporation, USA
Thomas E. Potok, Oak Ridge National Laboratory, USA
Christian Prehofer, Fraunhofer-Einrichtung für Systeme der Kommunikationstechnik ESK, Germany
Ela Pustułka-Hunt, Bundesamt für Statistik, Neuchâtel, Switzerland
Mengyu Qiao, South Dakota School of Mines and Technology, USA
Kornelije Rabuzin, University of Zagreb, Croatia
J. Javier Rainer Granados, Universidad Politécnica de Madrid, Spain
Muthu Ramachandran, Leeds Metropolitan University, UK
Thurasamy Ramayah, Universiti Sains Malaysia, Malaysia
Prakash Ranganathan, University of North Dakota, USA
José Raúl Romero, University of Córdoba, Spain
Henrique Rebêlo, Federal University of Pernambuco, Brazil
Hassan Reza, UND Aerospace, USA
Elvinia Riccobene, Università degli Studi di Milano, Italy
Daniel Riesco, Universidad Nacional de San Luis, Argentina
Mathieu Roche, LIRMM / CNRS / Univ. Montpellier 2, France
José Rouillard, University of Lille, France
Siegfried Rouvrais, TELECOM Bretagne, France
Claus-Peter Rückemann, Leibniz Universität Hannover / Westfälische Wilhelms-Universität Münster / North-German Supercomputing Alliance, Germany
Djamel Sadok, Universidade Federal de Pernambuco, Brazil
Ismael Sanz, Universitat Jaume I, Spain
M. Saravanan, Ericsson India Pvt. Ltd -Tamil Nadu, India
Idrissa Sarr, University of Cheikh Anta Diop, Dakar, Senegal / University of Quebec, Canada
Patrizia Scandurra, University of Bergamo, Italy
Daniel Schall, Vienna University of Technology, Austria
Rainer Schmidt, Munich University of Applied Sciences, Germany
Sebastian Senge, TU Dortmund, Germany
Isabel Seruca, Universidade Portucalense - Porto, Portugal

CONTENTS

An Integrated Model for Content Management, Topic-oriented User Segmentation, and Behavioral Targeting

Hans-Werner Sehring, Namics, Deutschland

Possibilities of the Reverse Run of Software Systems Modeled by Petri Nets

Radek Kočí and Vladimír Janoušek

Brno University of Technology, Faculty of Information Technology,
IT4Innovations Centre of Excellence
Bozetechova 2, 612 66 Brno, Czech Republic
{koci,janousek}@fit.vutbr.cz

Abstract—Application run tracing and step-by-step execution are an integral part of the debugging process. In many cases, debugging would be more comfortable and faster if it was possible to go back in the execution and examine the system state before getting into the wrong or disabled state. Currently, such a technique is not widespread, but there are experimental implementations that burden the application run with logging information needed to restore previous states. Moreover, many of them increase overhead in a significant way. This paper focuses on the possibility of reversing the run of systems whose behavior is described by Petri nets. The work follows the methodology of designing and validating system requirements using functional models that combine formal notation with objects of the production environment and can be used as a full-fledged application. Due to the nature of Petri Nets formalisms, it is possible to define reverse operations to reduce the overhead of application run.

Keywords–Object Oriented Petri Nets; debugging; tracing; reverse debugging; requirements validation.

I. Introduction

This work builds on the concepts of formal approach to design and develop system requirements and evolves principles discussed in the paper [1]. The presented approach uses the formalism of Petri Nets to specify the system under development [2]. It is part of the *Simulation Driven Development* (SDD) approach [3] combining basic models of the most used modeling language Unified Modeling Language (UML) [4][5] and the formalism of Object-Oriented Petri Nets (OOPN) [6]. This approach is based on ideas of model-driven development dealing with gaps between different development stages and focuses on the usage of conceptual models during the development process of simulation models. These techniques are called *model continuity* [7]. The model continuity concept works with simulation models during design stages, while the approach based on Petri Nets focuses on *live models* that can be used in the deployed system.

When testing models or implementations, developers often use the interactive debugging technique, which allows them to go through the system run and investigate its state step by step. The logging technique and subsequent analysis of the running system are less often used. These techniques are linked to the limits of their use, notably the inability to make reverse steps. In this case, it is difficult to determine the system states before stopping (e.g., at breakpoints). However, the introduction of reverse interactive debugging leads to increased overhead, especially for running an application where it is necessary to collect the information needed to reconstruct the previous states. There are several approaches; however, they differ in their possibilities and overhead. A significant factor is, in addition to higher demands on the runtime of the application, that there is a higher demand for memory that keeps the collected information. Another issue is the overhead of reverse debugging, which is not as important as the run overhead.

There are three basic approaches to solving this problem. The first one records the system run and then performs all the steps from the beginning to the desired point (*record-replay* approach). The second approach records all the information needed to return to the previous step (*trace-based* approach). The third approach records only selected checkpoints, so they are reliably replicated (*reconstruction-based* approach). Reverse debugging is done by reconstructing the appropriate checkpoint state and then making forward steps. The approach presented in this paper is based on the trace-based reverse debugging. Due to the nature of used OOPN formalism, which has a formal base working with uniquely defined events, there is no problem in defining and performing reverse operations associated with each event.

The paper is organized as follows. Section II introduces related work. Section III summarizes basic definitions of OOPN formalism needed to define tracing concepts. Section IV discusses the possibilities of OOPN models simulation tracing and introduces the simple demonstrating model. Section V focuses on recording states and event during the simulation and Section VI describes reverse events and operations when reverse debugging performed. The question of saving the whole state is discussed in Section VII. The summary and future work are described in Section VIII.

II. Related Work

The solution based on recording simulation run and replaying it from the beginning to the breakpoint may be time-consuming and, for a long run of the application, unsuitable due to time lags when debugging. As examples, we can mention Instant Replay debugger [8] or Microsoft Visual Studio 2010 IntelliTrace [9].

The trace-based solution logs all steps, so it is possible to determine the current state and the sequence of steps that led to this state. In many cases, the simulators record everything and, therefore, it is possible to go back to one of the previous

steps. The scope of that solution is limited by what and how can be traced, especially using multi-processors is very difficult to work. As examples, we can mention Green Hills Time Machine [10], Omniscient Debugger [11] fo Java Virtual Machine, or gnu reverse debugger gdb 7.0 [12]. The last mentioned, gdb debugger, is very slow but is the only open-source solution. There are tools based on Petri nets that allow reverse debugging, e.g., the TIme petri Net Analyzer TINA [13]. Nevertheless, these tools focus on a specific variant of Petri nets that are not usable for the application environment. Besides, there are also tools suitable for these purposes, e.g., Renew [14], which are similar to the SDD approach but do not allow reverse debugging.

Some solutions allow us to go back within the operation stack, change the current state, and proceed from this step. An example may be the Smalltalk language [15]. Even in this case, however, we do not have a stack of states associated with the appropriate steps. We only get the current state, which is reflected in all previously executed steps.

III. BASIC DEFINITION OF OOPN FORMALISM

In this section, we introduce the basic definition of Object-oriented Petri Nets (OOPN) formalism necessary for the presented purpose.

A. System of Classes and Objects

For this work, we define the Object Oriented Petri Nets (OOPN) as a system of classes and objects that consists of the individual elements [16].

Definition 1: System OOPN is $\Pi = (\Sigma, \Gamma, c_0, o_0)$, where Σ is a system of classes, Γ is a system of objects, c_0 is an initial class and o_0 is an identifier of the initial object instantiated from the class c_0.

Definition 2: System of classes Σ consists of sets of elements constituting classes and is defined as $\Sigma = (C_\Sigma, MSG, N_O, N_M, SP, NP, P, T, CONST, VAR)$, where C_Σ is a set of classes, MSG is a set of messages, N_O is a set of object nets, N_M is a set of method nets, SP is a set of synchronous ports, NP is a set of negative predicates, P is a set of places, T is a set of transitions, CONST is a set of constants and VAR is a set of variables. Messages MSG correspond to method nets, synchronous ports, and negative predicates.

Definition 3: The class is a structure

$$C = (MSG^C, n_O^C, N_M^C, SP^C, NP^C),$$

where $MSG^C \subseteq MSG$, $n_O^C \in N_O$, $N_M^C \subseteq N_M$, $SP^C \subseteq SP$ and $NP^C \subseteq NP$. Each net of the set $\{n_O^C\} \cup N_M^C$ consists of places and transitions (subsets of P and T).

Definition 4: System of objects Γ is a structure containing sets of elements constituting the model runs (the model run corresponds to the simulation, so that we will use the notation of simulation). $\Gamma = (O_\Gamma, N_\Gamma, M_N, M_T)$, where O_Γ is a set of object identifiers, N_Γ is a set of method nets identifiers, $M_N \subset (O_\Gamma \cup N_\Gamma) \times P \times U^M$ is place markings and $M_T \subset (O_\Gamma \cup N_\Gamma) \times T \times \mathcal{P}(BIND)$ is transition markings.

Definition 5: The OOPN system universe U is defined $U = \{(cnst, cls, oid) \mid cnst \in CONST \wedge cls \in C_\Sigma \wedge oid \in O_\Gamma\}$. The system universe represents a set of all possible values that may be part of markings or variables. $U^\mathcal{M}$ is then a multi-set over the set U.

We can use the following notation to simplify writing. For constants, we write down their values directly, e.g., 10, $'a'$. For classes, we write down their names directly without quotes or apostrophes. To identify an object, we write its identifier with a @ character.

Definition 6: The set of all variable bindings BIND used in OOPN is defined $BIND = \{b \mid b : VAR \longrightarrow U\}$.

Definition 7: We define operators for instantiating classes II_C and method nets II_N that create the appropriate instances and assign them identifiers from sets O_Γ, resp. N_Γ. When creating a new instance of the class $c \in C_\Sigma$, we will write $II_C(c) = o$ or $II_C(c, o)$, where $o \in O_\Gamma$. Similarly, for the method net instance $m \in N_M$, we will write $II_N(o, m) = n$ or $II_N(o, m, n)$, where $o \in O_\Gamma$ is an object where the method net instance $n \in N_\Gamma$ is created.

Individual class elements are identified by their fully qualified names consisting of sub-element names separated by a dot. The class is identified by its name, e.g., C. The method is identified by class and method names, e.g., C.M, the method place C.M.P, and so on. In the case of object net, the elements will be written directly without method identification, e.g., C.P. Similarly, we introduce the identification of Γ object system elements. Objects and nets instances are uniquely identified by their identifiers, net elements (transitions and places) by their names. For instance, the transition $t \in T$ of the method net $m_i \in N_\Gamma$ can be identified by following notations: $m_i.t$ or (m_i, t). The object net describes the autonomous activities of the object; its instance is always created with the instantiation of the class, and is just one. For this reason, the notation $o \in O_\Gamma$ can identify the class instance as well as its object net. Method nets describe the object's response to the sent message. In case the message is received, the instance $n \in N_\Gamma$ of the respective net N_M is created, and its simulation starts.

B. OOPN Model Example

An example illustrating the basic elements of OOPN formalism is shown in Figure 1. There are defined two classes C0 and C1, where the class C0 is marked as the initial one. When you run the simulation, an instance of the initial class is created, that is, the class C0. The class C0 has only an object net that contains places p1, p2, p3, p4 and transitions t1, t2, t3. These transitions gradually create instances of the class C1 in their actions and send doFor:20 and doFor:10 messages to these objects. The object net C1 contains the place p11 with an initial marking 10 and the method net doFor:. The method net has a place x corresponding to the x argument of the method, an output place return, where the resulting method token is placed, and two transitions t1 and t2.

Transitions are protected by guards. The transition t1 is fireable (can be executed, i.e., fired) if the value of the passed argument x is less than or equal to the value stored in the place p11 of the object net (x <= s). The transition t2 is
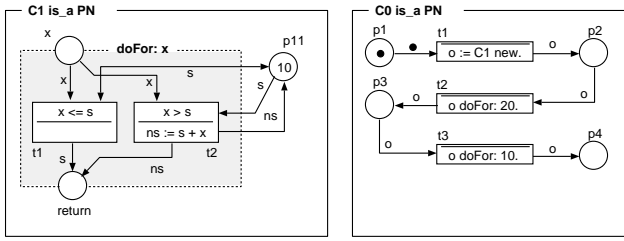
Figure 1.   An example of the OOPN model with basic elements.

fireable if the value of the passed argument x is greater than the value stored in the place p11 of the object net (x > s). Firing the transition t1 does not change the state of the object, and a copy of the value from the place p11 is put to the output place return. It indicates that the method has reached a result, and its net terminates when the calling transition obtains this result. By firing the transition t2, the value of the variable x is added to the value stored in place p11.

C. Nets of the OOPN Formalism

The OOPN class consists of two categories of nets, *object net* and *method net*. These nets are instantiated, i.e., their copy is created over which a simulation is performed. When the new object is being created, its object net is instantiated immediately. When the method is invoked, its net is instantiated. Now let us take a look at the example in Figure 1 and explain the model dynamics, i.e., the use of individual nets.
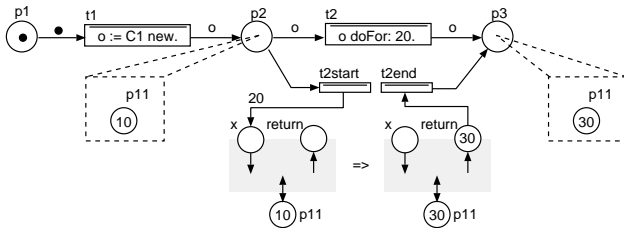
Figure 2.   A usage of the method net doFor: of the object net C0.

As soon as the simulation starts, an instance of the initial class (i.e., the class C0) is created. Part of the object net of the initial object o_0 is shown in Figure 2. First, the transition t1 is performed, which instantiates the class C1 class and puts a newly created object to the place p2. In place p2, there is now an object o_1 with an object net containing only one place p11 having the initial marking 10. Next, the transition t2 is fired, which sends a message doFor: with the argument value 20 taken from the place p2. We can imagine this situation as replacing the transition t2 with two auxiliary transitions t2start, respectively t2end. These transitions are associated with input, respectively output, places of the method net its instance was created. The object o_1 now contains an object net (constituted by the place p11) and an instance of the method net doFor:. When the method net completes the execution, the resulting token is placed to the place return. The auxiliary transition t2end fetches this token from the place return and ends the method net's run. An object o_1 is inserted to the place p3, which now contains only the object net with the place p11 containing the value of 30.

D. Place

The place is represented by a named multi-set. The multi-set A^M is a generalization of the set A such that it can contain multiple occurrences of elements. Thus, the multi-set can be defined as a function $A^M : A \rightarrow \mathbb{N}$, which assigns to each element $a \in A$ the number of occurrences in the multi-set. The number of occurrences will be denoted by the term *frequency*. We denote $|A|$ the cardinality of the set A, i.e., the number of elements in the set A. We denote $|A^M|$ the cardinality of multi-set A^M, i.e., the sum of frequencies of all elements in the set A. For an individual element x of the place $p \in P$, we write $x \in p$ a for its frequency m'x.

Definition 8: The place marking corresponds to its content and is defined as a multi-set $M_P = \{(m, o) \mid m \in \mathbb{N}^+ \wedge o \in U\}$, where m is frequency of the member o in the multi-set. Members of multi-set will be written in the form m'o, marking of the place $p \in P$ will be written in the form $M_P(p) = \{m_1'o_1, m_2'o_2, \dots\}$.

As an example we can mention the initial marking of the place $o_0.p1$ from the model shown in Figure 1—$M_P(o_0.p1) = \{1'\bullet\}$. The place cardinality is $|o_0.m_1.p2| = 1$ a $|o_0.m_1.p2^M| = 1$.

E. Transition

The transition is a net element whose execution determines the model's dynamics. OOPN dynamics are based on High-level Petri nets, but the semantics of transitions is modified. Each transition includes edges that connect the transition to places; the edge defines one condition for the fireability of the transition. The condition is defined by the inscription language. We distinguish input conditions (*precondition*), test conditions (*condition*) and output conditions (*postcondition*). The transition is fireable for certain *binding* of variables that are defined in the input and test conditions and the transition guard, if there are a sufficient number of objects in input places and the guard is evaluated truthfully for the obtained binding.

Individual types of transitions are defined as follows

- *Input condition* (*precondition*) associates a transition with a place whose state is a condition for firing (performing) the transition. When transition fires, the associated objects are removed from the associated input place. An example is the edge from place C0.p1 to the transition C0.t1.

- *Test condition* (*condition*) is similar to the input condition, except it does not change the state of the associated input place when the transition fires. The demonstration example does not include a test edge.

- *Output condition* (*postcondition*) specifies what objects are inserted in the associated output place after the transition fires. An example is the edge from the transition C0.t1 to the place C0.p2.

F. Inscription Language and Arc Expressions

An important part of OOPN formalism is its language of instruction, so-called *inscription language*. The language includes edge evaluation and operations defined in guards

and actions of transitions. The form of the script language is inspired by the Smalltalk language.

Arc expression matches the usual approach used in Petri nets. Each arc expression has a form of m'o, where $m \in \mathbb{N}^+ \cup VAR$ and $o \in U \cup VAR$. The expression element m represents the frequency of o in the multi-set and can be denoted by a numeric value or a variable. If the variable is used at the position m, the frequency of the member o in multi-set is assigned to that variable. The element o represents the object stored in multi-set and can be defined by the element of the universe U or the variable. If a variable is used at the position o, an object from multi-set, whose frequency corresponds to specified m, is bound to that variable. If both parts of an expression are defined by variables, any object and its frequency are bound to these variables. If the content of the multi-set does not match the given expression, the bounding process fails.

G. Synchronization Mechanisms

Synchronization elements and dynamics of the OOPN model are presented in the modified model, which is shown in Figure 3. Two classes C0 and C1 are defined here. Object net of the class C0 contains C0.p1 and C0.p2 and one transition C0.t1. The object net C1 is empty. The class C0 contains the method C1.init:, the synchronous port C0.get: and the negative predicate C0.empty. The class C1 contains the method C1.doFor:.

We show an example of calling synchronous port C0.get: with a free parameter. The synchronous port has one parameter named o. This port is called from the transition C1.t2 with free variable n. The parameter o is free, too, and is bound to one object from the place C0.p2. The bound object is available in the calling transition via the variable n. If the called synchronous port does not exist, the transition is not fireable.

An example of a negative predicate is C0.empty. The predicate is called from the transition C1.t3, which means that the transition C1.t3 is fireable if the place C0.p2 is empty. Negative predicate C0.empty is complementary to the synchronous port C0.get: called with a free parameter. Either one or the other is fireable, never both or none at the same time. In this way, it is possible to model a decision based on the existence of tokens in places.
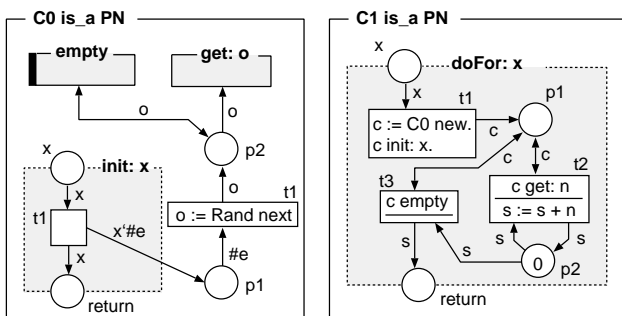
Figure 3. OOPN model with synchronization mechanisms.

Now let us look at the dynamics of the model that is created by instantiating the class C1, $\Pi_C(C1) = o_0$, and

calling the method doFor: 3 on the object o_0. When the method net is being instantiated, $\Pi_N(o_0, C1.doFor:) = m_1$, and a constant object 3 is inserted into the parameter place $o_0.m_1.x$ the transition $o_0.m_1.t1$ fires. Executing this creates an instance of the class C0, $\Pi_C(C0) = o_1$, a reference to object o_1 is stored in the variable c and object o_1 is initialized by calling the method $o_1.init$, $\Pi_N(o_1, C0.init) = m_2$. It inserts three symbols #e to the place $o_1.p1$. The transition $o_1.t1$ generates three random numbers and puts them in the place $o_1.p2$. Transitions $o_0.m_1.t2$ and $o_0.m_1.t3$ are complementary. The transition $o_0.m_1.t2$ is fireable if there is at least one object (number) in the place $o_1.p2$. Test of fireability and obtaining objects at the same time, if the test is successful, are provided by the synchronous port $o_1.get:$ with the free variable n. When the transition is performed, one object from the place $o_1.p1$ is bound to the variable n. At the same time, the bound port get: is executed on the object o_1 and the bound object is removed from the place $o_1.p2$. Transition $o_0.m_1.t2$ then adds the acquired value to the amount stored in the place $o_0.m_1.p2$. Figure 4 shows the principle of dynamic fusion of transition and synchronous port. Figure 4a) is the state after execution the transition $o_0.m_1.t1$. The place $o_0.m_1.p1$ contains a reference to the object o_1 and the place $o_1.p2$ contains three randomly generated values. Dynamic fusion of the transition $o_0.m_1.t2$ with synchronous port is shown in Figure 4b). After the transition $o_0.m_1.t2$ fires for the binding $xn = 10$, the model moves to the state shown in Figure 4c).
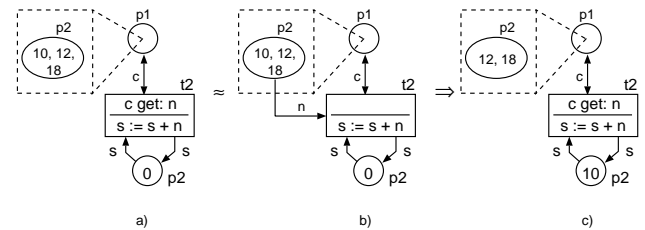
Figure 4. Dynamic fusion of the transition C1.doFor:.t2 and synchronous port C0.get:.

The transition $o_0.m_1.t3$ is fireable if the place $o_1.p2$ is empty. The fireability test is provided by calling the negative predicate $o_1.empty$. If the transition $o_0.m_1.t3$ fires, the value is transferred from the place $o_0.m_1.p2$ to the place $o_0.m_1.return$ and the method execution is terminated. Calling and executing the method $o_1.doFor:$ x generates x random numbers and returns the sum of them.

H. Set of Classes

The formalism of OOPN works, in addition to the OOPN objects (O_Γ and the corresponding set of classes C_Σ), with objects that are not a direct part of the formalism. The principle of their usage is based on Smalltalk, which is also used as the inscription language of the formalism of OOPN. These objects are especially *basic constant objects* (sometimes also called *primitive objects*) such as numbers, symbols, characters, and strings. The corresponding classes will be denoted Number, Symbol, Character and String and their set, in sum, C_C. Objects of these classes are part of the set of constants CONST. In addition to these basic objects, OOPN formalism

can work with other objects and classes. In particular, it covers collections, graphical user interface objects, user-defined classes, etc. We will call the set of these classes as *domain classes* and denote with the symbol $C_\mathcal{D}$, $C_\mathcal{C} \subset C_\mathcal{D}$. A set of object identifiers created from classes $C_\mathcal{D}$ is denoted $O_\mathcal{D}$.

Definition 9: Let $C_\Pi = C_\Sigma \cup C_\mathcal{D}$ be a set of all classes that can be used by the formalism of OOPN. Let $O_\Pi = O_\Gamma \cup O_\mathcal{D}$ be a set of all object identifiers that can be instantiated (created) from classes C_Π.

Definition 10: Extended Universe U_Π of OOPN is defined $U_\Pi = \{(\mathrm{cnst}, \mathrm{cls}, \mathrm{oid}) \mid \mathrm{cnst} \in \mathrm{CONST} \wedge \mathrm{cls} \in C_\Pi \wedge \mathrm{oid} \in O_\Pi\}$.

Definition 11: The set of selectors MSG can be defined as the union of the following sets: $\mathrm{MSG} = \mathrm{MSG_M} \cup \mathrm{MSG_S} \cup \mathrm{MSG_P} \cup \mathrm{MSG_D}$, where $\mathrm{MSG_M}$ corresponds to method nets, $\mathrm{MSG_S}$ corresponds to synchronous ports, $\mathrm{MSG_P}$ corresponds to predicates and $\mathrm{MSG_D}$ corresponds to domain class operations.

IV. SIMULATION TRACING

In this section, we briefly outline the sample model and discuss the possibilities of tracing the run of the software system described by the formalism of OOPN. We call that run *simulation*.

A. Sample Model

The basic concept will be outlined using a simple example. Figure 5 shows classes of that example. Figure 5a) depicts the initial class A1 with its object net and Figure 5b) depicts the class A2 having the only method calc: with one parameter x.
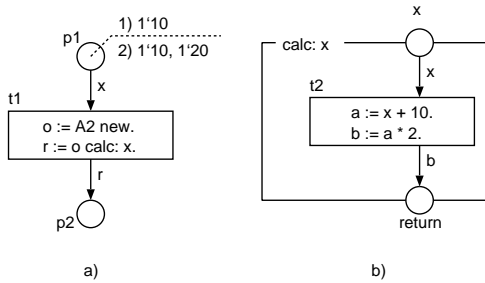
Figure 5. The sample model consisting of two classes A1 and A2; a) class A1 has only object net and b) class A2 has only method net calc:.

At the simulation start, an instance o_0 from the initial class A1 is created, $\Pi_\mathrm{C}(A1, o_0)$. The object net o_0 creates an instance of the class A2, $\Pi_\mathrm{C}(A2, o_1)$, and calls its method net calc: from the transition $o_0.t1$, $\Pi_\mathrm{N}(o_1, \mathrm{calc:}, n_1)$. The method net n_1 executes the transition $n_1.t2$. This example works with two variants of initial marking of the object net A1; 1) $M_\mathrm{P}(p1) = \{1\text{'}10\}$ and 2) $M_\mathrm{P}(p1) = \{1\text{'}10, 1\text{'}20\}$.

B. Tracing Tree

The simulation progress can be recorded as a tree, where nodes represent the relevant unit of simulation run, and edges represent a sequence of units execution, including the bindings. The relevant unit is understood as the least set of events that

the tracer records. Tree root represents the input point of the calculation. If a parallel calculation occurs during the execution of the relevant unit, this unit has more successors in its tree view. The current state of the calculation is then represented by all tree leaves. In Figure 6, we can see such a tree for the model from Figure 5 for the variant of initial marking $M_\mathrm{P}(p1) = \{1\text{'}10, 1\text{'}20\}$. In this example, the relevant unit is one executed command. Edges are recorded with a full-line arrow. Nodes capture on which net and transition the command has been performed.
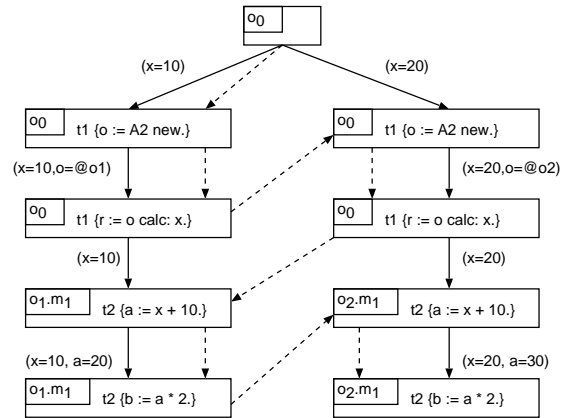
Figure 6. Scenario model of one simulation run.

The tree constructed by that way represents threads that may appear while running the simulation. It does not, however, capture the succession of steps that are important for making backward steps. The sequence of steps (events, see Section IV-C) can be different and depends on the specific conditions of the simulation run. One such variant is captured in Figure 6 with the dashed line arrows.

If all previous node states are stored during a trace, it is possible to return to any of the previous nodes at any time, make any changes to the state, and continue from that node. All nodes that are between the leaf and the selected node are removed from the tree. In this way, you can work independently on different tree branches. If there is a branch or link on the path, it must also be canceled. If another branch has been executed concurrently, it either stays (there is a connection on the way) or its execution must also be removed (there is a branch on the way).

C. Event

The simulation run is driven by events. Each executed (fired) event changes the system state, and, therefore, represents one step of model simulation. The set of states S of the system has a character of the net instances marking, which includes marking of places and transitions. One step from the state $s \in S$ to the state $s' \in S$ is written in the form $s\ [\mathrm{ev}\rangle\ s'$, where ev is an executed event.

Definition 12: Event is $\mathrm{ev} = (e, \mathrm{id}, t, b)$, where e is a type of event, $\mathrm{id} \in N_\Gamma \cup O_\Gamma$ is the identifier of net instance the event executes in, $t \in T$ is the transition to be executed (fired), and $b \in \mathcal{P}(\mathrm{BIND})$ is variables binding the event is to be executed for.

Event types can be as follows: A represents an atomic event, the entire transition is done in one step; F represents sending a message, i.e., creating an instance of a new method net and waiting for its completion; J represents completion of the method net called at F event.

(1) $\{m_1.p2\{1`0\}, m_1.x\{1`0\}\}$
$\Rightarrow [(F, m_1, C1.doFor:.t1, \{(x, 3)\})\rangle$

(2) $\{m_1.p2\{1`0\}, \}$
$\Rightarrow [(C, m_1, C1.doFor:.t1, \{(x, 3), (c, o_1)\})\rangle$

(3) $\{m_1.p2\{1`0\}, m_2.x\{1`3\}\}$
$\Rightarrow [(A, m_2, C0.init.t1, \{(x, 3)\})\rangle$

(4) $\{m_1.p2\{1`0\}, m_2.return\{1`3\}, o_1.p1\{3`\#e\}\}$
$\Rightarrow [(J, m_1, C1.doFor:.t1, \{(x, 3), (c, o_1)\})\rangle$

(5) $\{m_1.p2\{1`0\}, o_1.p1\{3`\#e\}, m_1.p1\{1`o_1\}\}$
$\Rightarrow [(A, m_2, C0.t1, \{(\varepsilon, \#e)\})\rangle$

(6) $\{m_1.p2\{1`0\}, o_1.p1\{2`\#e\}, o1.p2\{1`n_1\}, m_1.p1\{1`o_1\}\}$
$\Rightarrow [(A, m_1, C1.doFor:.t2, \{(c, o_1), (s, 0), (n, n_1)\})\rangle$

(7) $\{m_1.p2\{1`n_1\}, o_1.p1\{2`\#e\}, m_1.p1\{1`o_1\}\}$
$\Rightarrow \ldots$

Figure 7. An example of the transition sequence record.

Here are a few steps of the example that is modeled in Figure 3, and its dynamics are described in Section III-G. We start from a state where $\text{II}_N(o_0, C1.doFor:) = m_1$ is invoked with the parameter 3. Places that are empty in that state are not listed.

D. Event flow subgraph

From the object point of view, we distinguish an event whose execution does not require an external stimulus (*internal event*), and an event that comes from the outside (*external event*). A feasible transition always represents the internal event. An external event can be fired by sending a message or by calling a synchronous port. The message creates a net instance consisting of an internal event sequence. Synchronous port can affect object state by modification of places content, but also invoke the sequence of internal events in the object net.

The object net can describe multiple scenarios, either interconnected or totally disjoint. The structure of each net is defined by a graph of the Petri net, so we can define the scenarios as subgraphs of such nets.

Definition 13: Let $\mathcal{S}(O_\Gamma \cup N_\Gamma)$ be a set of all valid subgraphs of object nets O_Γ and method nets N_Γ. Individual scenarios will be denoted $\delta_c(n) = (ev_0, ev_1, \ldots)$, where $n \in O_\Gamma \wedge c \in \mathbb{N}$.

Now, we return to the step (i.e., event) sequence entry shown in Figure 6 and write the presented scenario in the form of net subgraph,

$$\delta = ([A, o_0, t1, (x = 10)\rangle, [F, o_0, t1, (x = 10, o = @o_1)\rangle,$$

$$[A, o_0, t1, (x = 20)\rangle, [F, o_0, t1, (x = 20, o = @o_2)\rangle,$$

$$[A, o_1.m_1, t2, (x = 10)\rangle, [A, o_1.m_1, t2, (x = 10, a = 20)\rangle,$$

$$[A, o_2.m_1, t2, (x = 20)\rangle, [A, o_2.m_1, t2, (x = 20, a = 30)\rangle)$$

Now, let us go back to our example and the event $[(A, m_1, C1.doFor:.t2, \{(c, o_1), (s, 0), (n, n_1)\})\rangle$. When this internal event is executed, the synchronized port $(o_1, C0.get:)$ is executed simultaneously, so the net o_1 will trigger an external event C0.get:, which removes one object from the place $(o_1, C0.p2)$. Execution of the net m_1 corresponds to a sequence of internal events (in abbreviated notation)

$$\delta_1(m_1) = ([(F, t1)\rangle, [(C, t1)\rangle, [(J, t1)\rangle, [(A, t2)\rangle,$$

$$[(A, t2)\rangle, [(A, t2)\rangle, [(A, t3)\rangle)$$

and execution of the net o_1 corresponds to the sequence of events (in abbreviated notation, external event is highlighted by superscript [e])

$$\delta_1(o_1) = ([(A, t1)\rangle, [(A, t1)\rangle, [(A, t1)\rangle,$$

$$[(A, get:)\rangle^e, [(A, get:)\rangle^e, [(A, get:)\rangle^e.$$

For the sake of readability, it is possible not to include in the notation those components which are not important for the described situation. In our case, we only show the type of event and the transition, in other situations a simple sequence of transitions is sufficient.

E. Composite Command

If the transition contains a sequence of messages, either step-by-step or composite ones, this transition can be understood, from the OOPN theory point of view, as a sequence of simple transitions, each of which contains just one simple command. An example of such equivalence is shown in Figure 8.
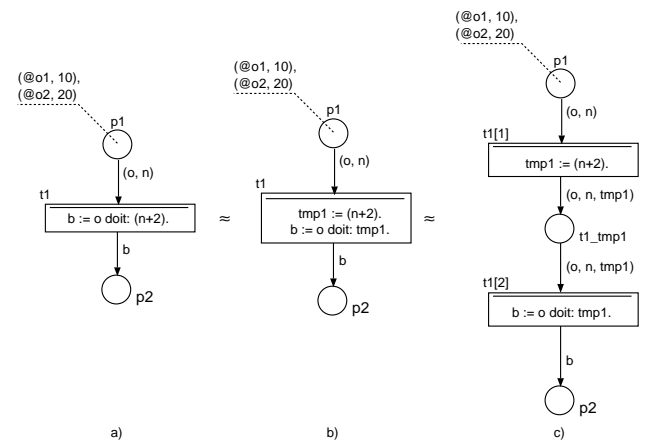
Figure 8. Composite command of the transition.

This model has four variants of execution. In the following example, only one is listed, the others are a combination of different interleaving of two concurrently running transitions

t1. The notation of a transition using index, e.g., $t1[1]$, refers to the corresponding command of the composite transition.

$$\delta_1(o_0) = ([A, o_0, t1[1], \{o = @o1, n = 10\}\rangle,$$
$$[F, o_0, t1[2], \{o = @o1, n = 10, tmp1 = 12\}\rangle,$$
$$[J, o_0, t1[2], \{(o = @o1, n = 10, tmp1 = 12\}\rangle,$$
$$[A, o_0, t1[1], \{(o = @o2, n = 20\}\rangle,$$
$$[F, o_0, t1[2], \{o = @o2, n = 20, tmp1 = 22\}\rangle,$$
$$[J, o_0, t1[2], \{o = @o2, n = 20, tmp1 = 22\}\rangle)$$

Due to that reason we slightly modify the transition marking definition, $M_T \subset (O_\Gamma \cup N_\Gamma) \times T \times \mathbb{N} \times \mathcal{P}(\text{BIND})$, where $\mathbb{N}$ represents an index of the composite transition command.

V. Recording the Simulation

This section focuses on recording states during simulation. We describe each of the monitored events and how the state changes are recorded. To record the entire state would be very time consuming and memory intensive, and from the means offered by OOPN formalism point of view also unnecessary. For stepping, it is sufficient to save partial state changes that avoid storing the whole simulation image after every step.

A. State Changes Processing

A partial state change may involve inserting or selecting an element from a place, assigning the result to a variable, creating or destroying an object, creating or completing a method net instance (associated with calling and terminating this method), and creating or completing a transition instance.

1) Changing Place State: Changing the place state is the easiest operation corresponding to removing elements when transition fires or adding elements when the transition is complete. Within one step, more elements can be inserted or removed into or out of more places. The change is recorded in the following notation. We define the operation $\text{add}(p, m, o)$ for adding element to the place and operation $\text{del}(p, m, o)$ for removing element from the place. In both operations, $p \in P$ is the place, and $m \in \mathbb{N}^+$ is the frequency of element $o \in U$.

2) Firing and Completing Composite Transition: Although the composite command in the transition is always interpreted as a sequence of individual commands, it is necessary to maintain a relationship with the original entire transition. Additionally, the transition can be run multiple times for different bindings, so it is necessary to identify the specific transition instance uniquely. Therefore, we introduce a particular event type B, which represents the transition firing for a given binding, and at the same time, assigns a unique identifier to the fired transition. Similarly, we introduce a particular C event type to completing the fired transition.

Definition 14: For writing state changes, we extend the definition of the system of objects Γ to the set of transition instance identifiers T_Γ, i.e., transitions fired with a specific binding, $\Gamma = (O_\Gamma, N_\Gamma, T_\Gamma, M_N, M_T)$.

3) Changing Variable State: Changing the state of the variable when executing the transition is denoted by operation $\text{swap}(t_i, v, o_{new}, o_{old})$, where $t_i \in T_\Gamma$ is a transition, $v \in \text{VAR}$ is the transition variable, $o_{new} \in U$ is a universe object assigned to the variable v and $o_{old} \in U$ is the original object assigned to the variable v before this event occurs.

4) Creating Object: Creating an object (a class instance) corresponds to the creation of an object net and its initialization. In terms of state recording, it is essential to keep information about the identification of newly created object II_C and changes of the object net's places, i.e., adding objects into places during the net initialization process.

5) Creating Method Net Instances: Creating a method net instance corresponds to a method invoking by sending a message. As with the object, it is necessary to keep information about the identification of newly created instance II_N and inserting objects (values) into the net's parameter places.

6) Completing Method Net Instances: After the method net instance is completed, two possible options can be applied to record changes. First, the current state of the entire net is recorded, i.e., marking of all places and all fired transitions (instances). Second, no state is recorded. The first option is more demanding for time and memory space during simulation. On the other side, it is not necessary to reconstruct the net's state so that it matches the state before its completion. The second option is more efficient during simulation, but it is more demanding to reconstruct the net's state during backward stepping. At this point, we focus on the option without state recording. We introduce a special operation $\Delta_N(m_i)$, which indicates the completion and cancellation of the net instance m_i.

B. Example of Tracing Simulation

We demonstrate the concept of simulation tracking on the model shown in Figure 5 for variant 1, i.e., with the initial marking $M_P(p1) = \{1`10\}$. For the reasons given in Section V-A2, we will modify the event definition as follows:

Definition 15: Event is $ev = (e, id, t, t_i, b)$, where e is a kind of event, $id \in N_\Gamma \cup O_\Gamma$ is the identifier of the net instance the event executes in, $t \in T$ is the transition to be executed (fired), and $b \in \mathcal{P}(\text{BIND})$ is variables binding for which this event is executed, and $t_i \in T_\Gamma$ is the identifier of fired transition.

There is only one variant of execution. The sequence of events is captured in Figure 9. For this text, we simplify writing so that we do not specify the binding b, and where it is evident from the context, we will not even specify the net identifier id.

$$\delta(o_1) = ([B, t1, t1_1\rangle, [A, t1[1], t1_1\rangle, [F, t1[2], t1_1\rangle, \delta(m_1),$$
$$[J, t1[2], t1_1\rangle, [C, t1, t1_1\rangle)$$
$$\delta(m_1) = ([B, t2, t2_1\rangle, [A, t2[1], t2_1\rangle, [A, t2[2], t2_1\rangle,$$
$$[J, t1, t1_1\rangle)$$

Figure 9. Scenario (event flow) of the sample example.

The sequence of fired transitions does not necessarily correspond to the tracing tree, which also takes into account the simulation branching. The sequence of fired transitions captures a specific sequence of events, which is always unambiguously given. Figure 10 captures the sequence of events (scenario) completed with state change operations. It represents

Event	State
	$\text{II}_C(\text{A1}, o_1)$
	$\text{add}(o_1.\text{p1}, 1, (10, \varepsilon, \varepsilon))$
$[\text{B}, o_1, \text{t1}, \text{t1}_1\rangle$	$\text{del}(o_1.\text{p1}, 1, (10, \varepsilon, \varepsilon))$
	$\text{swap}(\text{t1}_1, \text{x}, (10, \varepsilon, \varepsilon), \varepsilon)$
$[\text{A}, o_1, \text{t1}[1], \text{t1}_1\rangle$	$\text{II}_C(\text{A2}, o_2)$
	$\text{swap}(\text{t1}_1, \text{o}, (\varepsilon, \varepsilon, o_2), \varepsilon)$
$[\text{F}, o_1, \text{t1}[2], \text{t1}_1\rangle$	$\text{II}_N(o_2, \text{A2.calc:}, \text{m}_1)$
	$\text{add}(o_2.\text{m}_1.\text{x}, 1, (10, \varepsilon, \varepsilon))$
$[\text{B}, o_2.\text{m}_1, \text{t2}, \text{t2}_1\rangle$	$\text{del}(o_2.\text{m}_1.\text{x}, 1, (10, \varepsilon, \varepsilon))$
	$\text{swap}(\text{t2}_1, \text{x}, (10, \varepsilon, \varepsilon), \varepsilon)$
$[\text{A}, o_2.\text{m}_1, \text{t2}[1], \text{t2}_1\rangle$	$\text{swap}(\text{t2}_1, \text{a}, (20, \varepsilon, \varepsilon), \varepsilon)$
$[\text{A}, o_2.\text{m}_1, \text{t2}[2], \text{t2}_1\rangle$	$\text{swap}(\text{t2}_1, \text{b}, (40, \varepsilon, \varepsilon), \varepsilon)$
$[\text{C}, o_2.\text{m}_1, \text{t2}, \text{t2}_1\rangle$	$\text{add}(o_2.\text{m}_1.\text{return}, 1, (40, \varepsilon, \varepsilon))$
$[\text{J}, o_1, \text{t1}[2], \text{t1}_1\rangle$	$\Delta_N(\text{m}_1)$
	$\text{swap}(\text{t1}_1, \text{r}, (40, \varepsilon, \varepsilon), \varepsilon)$
$[\text{C}, o_1, \text{t1}_1\rangle$	$\text{add}(o_1.\text{p2}, 1, (40, \varepsilon, \varepsilon))$

Figure 10. Scenario record.

a tracing simulation with storing relevant information for backward stepping. We can see state changes in the *State* column. For this text, we simplify writing events so that we do not specify the binding b.

VI. REVERSE DEBUGGING

In this section, we describe steps that are performed when stepping backwards.

A. State Changes Reverse Processing

There is a sequence of reverse operations for each state change that allows to return to the previous step. We explain the operations associated with each recorded event. Some of the operations are demonstrated on the discussed example, the first steps of reverse debugging are shown in Figure 11.

1) C-Event Type: The event C represents completing the transition instance t_i. In a step back, our goal is to reconstruct this instance. It is necessary to perform the reverse operations that are associated with this event. Since these operations refer to the insertion of elements into the output places, the reverse operations remove these elements. The next step is to reconstruct the state of transition instance t_i. We find the first entry regarding the instance t_i, i.e., $[\text{B}, \text{t}, \text{t}_i\rangle$, create this instance and perform all the swap operations. In our example, this would be a sequence of events $[\text{B}, \text{t1}, \text{t1}_1\rangle$, $[\text{A}, \text{t1}[1], \text{t1}_1\rangle$, $[\text{F}, \text{t1}[2], \text{t1}_1\rangle$ and $[\text{J}, \text{t1}[2], \text{t1}_1\rangle$. Event B ensures creation of the appropriate instance with the t1_1 identifier. The associated sequence of swap operators is as follows: $\text{swap}(\text{t1}_1, \text{x}, (10, \varepsilon, \varepsilon), \varepsilon)$, $\text{swap}(\text{t1}_1, \text{o}, (\varepsilon, \varepsilon, o_2), \varepsilon)$ and $\text{swap}(\text{t1}_1, \text{r}, (40, \varepsilon, \varepsilon), \varepsilon)$. This way we filled all the variables with appropriate values, and we are in a state where the transition instance t1_1 was completed. If the object, resp. its identifier, that has been destroyed (e.g., because it was removed by a garbage collector) is assigned to the variable, it is not essential at this point. The object will

be reconstructed at the first access to it (state handling, work with method net, etc.).

2) J-Event Type: The event J represents completing the call of method. The reverse swap operation is executed, i.e., the value is removed from the variable and replaced with the original (previous) value. The next step is to perform a reverse operation $\bar{\Delta}_N(\text{m}_i)$ to destroying the method net $\Delta_N(\text{m}_i)$, i.e., creating net instance m_i and reconstructing its last state. Using operation $\bar{\Delta}_N(\text{m}_i)$, we get a sequence of operations over the net m_i starting with $\text{II}_N(o_i, \text{class.method\_name}, \text{m}_i)$ operation. From this sequence, we will perform add and del operations on the net instance m_i. In our example, it would be a sequence of operations $\text{add}(o_2.\text{m}_1.\text{x}, 1, (10, \varepsilon, \varepsilon))$, $\text{del}(o_2.\text{m}_1.\text{x}, 1, (10, \varepsilon, \varepsilon))$ and $\text{add}(o_2.\text{m}_1, \text{return}, 1, (40, \varepsilon, \varepsilon))$. As a result, we made method net in the state, where the place return contains the object representing number 40.

Step	State
$[\text{C}, o_1, \text{t1}_1\rangle$	$\text{del}(o_1.\text{p2}, 1, (40, \varepsilon, \varepsilon))$
$[\text{J}, o_1, \text{t1}[2], \text{t1}_1\rangle$	$\text{swap}(\text{t1}_1, \text{r}, \varepsilon, (40, \varepsilon, \varepsilon))$
	$\bar{\Delta}_N(\text{m}_1) \Rightarrow$
	$\quad \text{II}_N(o_2, \text{A2.calc:}, \text{m}_1)$
	$\quad \text{add}(o_2.\text{m}_1.\text{x}, 1, (10, \varepsilon, \varepsilon))$
	$\quad \text{del}(o_2.\text{m}_1.\text{x}, 1, (10, \varepsilon, \varepsilon))$
	$\quad \text{add}(o_2.\text{m}_1.\text{return}, 1, (40, \varepsilon, \varepsilon))$
$[\text{C}, o_2.\text{m}_1, \text{t2}, \text{t2}_1\rangle$	$\text{del}(o_2.\text{m}_1.\text{return}, 1, (40, \varepsilon, \varepsilon))$
	$\text{swap}(\text{t2}_1, \text{x}, (10, \varepsilon, \varepsilon), \varepsilon)$
	$\text{swap}(\text{t2}_1, \text{a}, (20, \varepsilon, \varepsilon), \varepsilon)$
	$\text{swap}(\text{t2}_1, \text{b}, (40, \varepsilon, \varepsilon), \varepsilon)$
$[\text{A}, o_2.\text{m}_1, \text{t2}[2], \text{t2}_1\rangle$	$\text{swap}(\text{t2}_1, \text{b}, \varepsilon, (40, \varepsilon, \varepsilon))$

Figure 11. Reverse scenario.

There may be still instances of transitions that are not terminated at the method net completion. It may happen, although it very probably indicates a mistake in the design, that there are still instances of transitions that are not terminated at the method net completion. These instances must also be reconstructed. From the sequence of operations $\bar{\Delta}_N(\text{m}_i)$, we find such sequences that correspond to unfinished transitions starting with $[\text{B}, \text{t}, \text{t}_i\rangle$ event, but having no event $[\text{C}, \text{t}_i\rangle$. For each such sequence we perform actions similarly to the backward step of $[\text{C}, \text{t}_i\rangle$ event.

3) F-Event Type: The event F represents the method invoking on the object. In the reverse step, the appropriate instance of method net specified in II_N operator is destroyed.

4) A-Event Type: The event A represents the atomic execution of the operation. The reverse swap operation is executed, i.e., the value is removed from the variable and replaced with the original (previous) value. If the atomic operation is a creation of a class instance II_C, this instance is destroyed.

5) B-Event Type: The event B represents the start of transition execution (creation of a transition instance). In the reverse step, the transition instance t_i is destroyed, and the add reverse operation is performed. There is no need to swap variables, as the entire fired transition is canceled.

B. Object Reconstruction

At the time of access to the object, e.g., due to the reconstruction of the method net, it may happen that the object no longer exists. The reason may be the loss of all references to this object and its removal by the garbage collector. At this point, it is necessary to create the object and reconstruct its last state. Because the object was destroyed, it means that there were no existing method nets. It is necessary to reconstruct the state of the object net, which is done in the same way as the method net reconstruction. The sequence of corresponding operations on the object net o_i is obtained by using $\bar{\Delta}_O(o_i)$ operation, which is similar to $\bar{\Delta}_N(o_i)$ operation, but the obtained sequence starts with $\amalg_C(class, o_i)$ operation and the class instance is created instead of method net instance.

VII. Saving the Simulation State

As already mentioned, one way of ensuring the consistency of the state in reverse debugging is by continuously saving the state. In this section, we outline the way of the object's state saving and the associated problems.

A. Simulation State

The problem associated with continually saving changes may be the time consuming to reconstruct a state that is far from the current location, i.e., the simulation state. The distance μ can be defined as the number of steps taken from the start of the simulation, or the last saved state, to the stop point of the simulation. From this perspective, it seems appropriate to provide continuous saving the entire state at specific distances (after a certain number of steps) and saving changes from that storage point. In this way, the process of reconstructing states according to debugging requirements can be accelerated. On the other hand, it increases the demands of both spatial (the need to store more data) and time during the simulation. The question arises as to the appropriate choice of distance μ, after which it is useful to save the whole state of the simulation.

Definition 16: Simulation state $\mathcal{S}$ corresponds to the system of objects Γ consisting of sets of nets identifiers, net markings, and transition markings, $\mathcal{S} = (\Gamma, R_T)$, where $R_T \subseteq N_\Gamma \times (T, \mathbb{N})$ is a relation mapping fired method nets to fired transitions from which the nets have been called.

B. Domain objects

The second, more serious problem is related to objects that are not directly managed by the simulation engine. These are primarily objects of the production environment $O_\mathcal{D}$, which are instances of the domain classes $C_\mathcal{D}$. In this case, it is not possible to reconstruct the state as presented in the previous sections (except for the constant objects $O_\mathcal{C}$).

We can divide domain objects into several categories and work with them accordingly when trying to save a simulation state:

- *General objects.* Their concrete form cannot be specified in any way and, thus, it is impossible to define some limitations on these objects. They cannot be easily managed.

- *Collection.* Collections can be understood as specific objects, and the simulator can be prepared for it; their storage and recovery can be ensured in a relatively simple manner. However, objects stored in collections fall into the general object category.

- *Interface to other systems.* These objects serve as wrappers for other objects and as such can be understood as constant objects. Therefore, these objects do not need to be stored. Again, the objects that the interface encapsulates are general objects.

- *Objects that are linked to the model* (that is, the model may not have references to them). These objects are mostly outside the simulation management and use model objects mainly to determine the state by which they set their own state. Typically, these are user interface objects. This group can be understood as a specific subset of general objects.

- *Simulated general objects.* These objects are mostly used instead of ordinary general objects to simulate their behavior and reactions. They are used when it is impossible to connect created models to the real environment, or it is not necessary.

C. General Domain Objects

General domain objects are a significant problem in providing state during the simulation. One solution is to use the resources of the production environment, i.e., the environment in which the domain objects are implemented, and to serialize and deserialize those objects. In many cases, however, this solution requires to modify domain class definitions, e.g., to use annotations associated with a suitable Java toolkit. But, such an approach is so demanding that it is unusable in practice. Another problem is the large object structures where an object referenced from the OOPN model contains references to other nested objects. Ensuring their consistency is a significant problem in the reconstruction of complex object structures.

The specific solution to this problem then depends on the environment in which the entire system for modeling and verifying requirements is implemented. The implementation in Smalltalk can use a relatively clear and straightforward means of storing the state of objects without having to interfere in any way with the existing domain object code [15]. In this case, the combination of continually saving state changes and saving the entire state after μ steps is applied. Generic domain objects can be stored with the entire structure, or the plunging depth can be limited.

It is necessary to take into account that when debugging with backward steps, it is not possible to ensure a complete reconstruction of all domain objects, but only the basic ones, as mentioned above. Additionally, some of the general domain objects are still in the last achieved state. Nevertheless, we have to realize that the system, for which the discussed backward stepping principle is intended, serves mainly for the analysis and verification of requirements, as presented in [17]. General domain objects are mainly used to link the created model to existing components for system validation under real conditions. Under these circumstances, it can be accepted if

some actions on the domain objects are not performed. It is still possible to trace back the location where the error occurred, or the wrong data was sent to the domain objects. If a simulated domain object is used for validation purposes, the situation at the state reconstruction is much simpler.

VIII. CONCLUSION

The paper dealt with the concept of tracing and reversing a run of software system modeled by Petri Nets, especially the formalism of Object Oriented Petri nets. While the model is running, the sequence of events and changes in states of individual elements are logged. The presented concept, combined with saving the execution state, is fully functional but has not yet taken into account all the possibilities of use. We have abstracted the possibility of having objects that have running method nets, even though the garbage collector collected them. The reason is the existence of cyclic dependencies but unavailable from the initial object. We also were only concerned with pure Petri Nets objects and passed the domain objects, e.g., collections, objects of user classes, etc. The way of storing the state of generic domain objects remains an open question. One possibility is to save these states only at predefined points and then reconstruct them on the fly - this would involve the actual steps taken during the reconstruction. In any case, each solution requires collaboration of the host environment or subordination of domain object management to the OOPN simulator.

At present, we have an experimental partial implementation of the tool supporting reverse debugging. We will complete the implementation in the future and focus on the limitation mentioned above. For more effective stepping, it may be necessary to introduce a reconstruction of the whole state at specific points (e.g., at breakpoints). To provide greater flexibility in interactive debugging, we will consider the possibility to define the granularity of the unit to be traced (complete transition, command, method) at runtime.

ACKNOWLEDGMENT

REFERENCES

[1] R. Kočí and V. Janoušek, "Tracing and Reversing the Run of Software Systems Implemented by Petri Nets," in ThinkMind ICSEA 2018, The Tvelfth International Conference on Software Engineering Advances. Xpert Publishing Services, 2018.

[2] R. Kočí and V. Janoušek, "Specification of Requirements Using Unified Modeling Language and Petri Nets," International Journal on Advances in Software, vol. 10, no. 12, 2017, pp. 121–131.

[3] R. Kočí and V. Janoušek, "Modeling and Simulation-Based Design Using Object-Oriented Petri Nets: A Case Study," in Proceeding of the International Workshop on Petri Nets and Software Engineering 2012, vol. 851. CEUR, 2012, pp. 253–266.

[4] J. Rumbaugh, I. Jacobson, and G. Booch, The Unified Modeling Language Reference Manual. Addison-Wesley, 1999.

[5] C. Raistrick, P. Francis, J. Wright, C. Carter, and I. Wilkie, Model Driven Architecture with Executable UML. Cambridge University Press, 2004.

[6] M. Češka, V. Janoušek, and T. Vojnar, "Modelling, Prototyping, and Verifying Concurrent and Distributed Applications Using Object-Oriented Petri Nets," Kybernetes: The International Journal of Systems and Cybernetics, vol. 2002, no. 9, 2002.

[7] D. Cetinkaya, A. V. Dai, and M. D. Seck, "Model continuity in discrete event simulation: A framework for model-driven development of simulation models," ACM Transactions on Modeling and Computer Simulation, vol. 25, no. 3, 2015.

[8] T. LeBlanc and J. Mellor-Crummey, "Debugging Parallel Programs with Instant Replay," IEEE Transactions on Computers, vol. 36, no. 4, 1987, pp. 471–482.

[9] I. Huff, "IntelliTrace in Visual Studion 2010 Ultimate," MSDN Blogs, http://blogs.msdn.com/b/ianhu/archive/2009/05/13/historical-debugging-in-visual-studio-team-system-2010.aspx, 2009.

[10] M. Lindahl, "The Device Software Engineer's Best Friend," in IEEE Computer, 2006.

[11] B. Lewis and M. Ducasse, "Using Events to Debug Java Programs Backwards in Time," in Proc. of the ACM SIGPLAN 2003 Conference on Object-oriented programming, systems, languages, and applications (OOPSLA), 2003, pp. 96–97.

[12] The GNU Project Debugger, "GDB and Reverse Debugging," GNU pages, https://www.gnu.org/software/gdb/news/reversible.html, 2009.

[13] F. V. B. Berthomieu, F. Peres, "Model-checking Bounded Prioriterized Time Petri Nets," in Proceedings of ATVA, 2007.

[14] O. Kummer, F. Wienberg, and et al., "Renew – User Guide," http://www.informatik.uni-hamburg.de/TGI/renew/renew.pdf, January 2016.

[15] A. GoldBerk and D. Robson, Smalltalk 80: The Language. Addison-Wesley, 1989.

[16] R. Kočí and V. Janoušek, "The Object Oriented Petri Net Component Model," in The Tenth International Conference on Software Engineering Advances. Xpert Publishing Services, 2015, pp. 309–315.

[17] R. Kočí and V. Janoušek, "Modeling System Requirements Using Use Cases and Petri Nets," in ThinkMind ICSEA 2016, The Eleventh International Conference on Software Engineering Advances. Xpert Publishing Services, 2016, pp. 160–165.

Database Model Visualization in Virtual Reality: Exploring WebVR and Native VR Concepts

Roy Oberhauser
Computer Science Dept.
Aalen University
Aalen, Germany
e-mail: roy.oberhauser@hs-aalen.de

Abstract - **Databases are becoming an ubiquitous and integral part of most software as the data era and the Internet of Everything unfolds. Alternative database types such as NoSQL grow in popularity and allow data to be stored and accessed more simply or in new ways. Thus, software developers, not just database specialists, are more likely to encounter and need to deal with databases. Virtual Reality (VR) technology has grown in popularity, yet its integration in the software development tool chain has been limited. One potential application area for VR technology that has not been sufficiently explored is database-model visualization. This paper describes Virtual Reality Immersion in Data Models (VRiDaM), a generic database-model approach for visualizing, navigating, and conveying database-model information interactively. It describes and explores both native VR and WebVR solution concepts, with prototypes showing the viability of the approach.**

Keywords - virtual reality; database visualization; database tools; visual development environments; database modeling; software engineering; WebVR; Benediktine space.

I.   INTRODUCTION

This article is an extended version of a former conference publication, see [1] for further details. In our time and the foreseeable future, data has become the most coveted "raw material", in some respects analogous to a gold rush. IDC estimates the current annual data creation rate at 16.1ZB (Zettabytes), and by 2025 163ZB, with embedded data by then constituting nearly 20% of all data created [2]. Cisco estimates there will be 27bn networked devices by 2021 [3]. The ongoing digitalization involving Industry 4.0 and the Internet of Everything will imply a large increase in databases to be able to store and retrieve this data, in particular embedded databases. As data explodes, software engineers are increasingly faced with the daunting task of structuring and analyzing such databases across various DataBase Management System (DBMS) types, including relational and NoSQL types such as document (semi-structured), key-value, wide column (extensible record), in memory, and graph [4].

Faced with developing and maintaining software that integrates some form of data store, software engineers must increasingly deal with analyzing and changing data models. While the original developers may have (had) a clear (and correct to a certain degree) mental model of their actual data model, the maintenance situation is exacerbated by proliferation of relational (mostly SQL) and NoSQL database

types. Furthermore, the relatively high turnover rates common in the software industry can result in developers unfamiliar with the data models attempting to quickly comprehend the database structures involved with these software applications.

With regard to comprehension, humans tend to be visually-oriented and can detect and remember visual patterns well. Information visualization has the potential to support human understanding and insight while dealing with resource constraints on the human as well as computer side. While common ways for visually conveying database structures include 2D entity-relationship (E-R) modeling and diagrams [5], these are typically applied to relational databases (RDB), while NoSQL databases may or may not provide a tool that includes visual support. Usually a database product will provide a preferred product-specific tool offering that may be web-based or have a standard 2D graphical user interface (GUI), whereas certain tools can support multiple database products of one specific type (e.g., MySQL Workbench for SQL databases).

Contemporaneously, VR provides new options and capabilities for visual immersion and has made inroads in its accessibility, as prices have dropped and software and hardware capabilities have improved. The VR market has been rapidly expanding, with VR revenue reaching $2.7bn in 2016, with an expected 10-fold increase to $25bn by 2021 [6]. Broad VR usage is still relatively new and the developer market segment small in comparison to the general VR market segment. Thus, software engineers as yet do not have access to integrated VR capabilities in their development tool chains. In this respect, the application of VR to database structures has been insufficiently explored, and a general approach and specific solutions concepts for utilizing native VR and WebVR are not yet prevalent.

This paper contributes Virtual Reality Immersion in Data Models (VRiDaM, pronounced like freedom), a generalized approach to heterogeneous (relational and non-relational) database-model visualization in VR. In this paper we explore two specific VR solution concepts: 1) WebVR in a web browser using a Benediktine-space [7]-[10] visualization paradigm, and 2) a native VR visualization concept. Database models from different data store types can be visualized and navigated (locally or remotely) in VR using a VR headset and controller. We describe principles and features for visualizing, navigating, and conveying database information interactively to support exploratory, analytical, and descriptive cognitive processes [11]. Prototype implementations demonstrate the

viability of the approach. This paper extends [1] by contributing a generalized approach and providing both native VR and WebVR solution concepts to address database visualization.

The paper is organized as follows: the following section discusses related work. Section III provides background information. Section IV presents our general solution approach and two solution concepts. Section V describes the implementation. In Section VI, the solution is evaluated and is followed by the conclusion.

II. RELATED WORK

While we found no directly related work, we did find work related to the application of VR for visualizing data in specific areas such as astronomy, biology, geography, etc. Okada et al. use VR to visualize and explore tweet data [12][13]. Moran et al. [14] developed a Unity3D application for geospatial visual analytics of Twitter data. Sun and Wu [15] describe a VR data analytics platform supporting multidimensional data in a geographic based view, specifically presenting factory chemical readings and meteorological data. While VR Juggler [16] provides VR support for developing VR applications, it does not address database modeling and visualization. As to VR approaches for software structure visualization, ExplorViz [17] is a WebVR application that supports VR exploration of 3D software cities using Oculus Rift together with Microsoft Kinect for gesture recognition. As to non-VR visualization, [18] provides an overview and survey of 3D software visualization tools across various software engineering areas. Software Galaxies [19] gives a web-based visualization of dependencies among popular package managers and supports flying, with each star representing a package clustered by dependencies. CodeCity [20] is a 3D software visualization approach based on a city metaphor and implemented in SmallTalk on the Moose reengineering framework. X3D-UML [21] provides 3D support with UML grouping classes in planes. In contrast, VRiDaM focuses on specifically on visualizing database structures.

Database management (DBM) tools are typically DB type-specific and require some installation. Each professional RDB product usually offers its own tool, but since most RDBs support the Structured Query Language (SQL), certain RDB tools can access other RDBs using RDB-specific drivers. For example, MySQL Workbench works across multiple databases and supports reverse-engineering of 2D E-R diagrams. Schemaball [22] provides a 2D circular composite schema diagram for SQL databases. As to 3D RDB tools, DIVA (Database Immersive Visual Analysis) uses stacked rings with rectangular tables attached to them (forming a cylinder), with the tables with the most foreign keys sorted to the top of the stack. Alternatively, stacked square layers of tables can be displayed and 2D E-R diagrams shown. Actual data values are not visualized. For NoSQL databases, each database type differs and the associated DBM tools. One example of a popular wide-column database (WDB) is Apache Cassandra, for which DataStax Studio is a Java-based DBM tool with a web GUI (Graphical User Interface). As to document-oriented databases (DDBs), MongoDB is a popular example and MongoDB Compass, Robomongo, and Studio

3T being example DBM tools. For graph databases (GDB), Neo4j is popular and graph DBM tools include Neo4j admin, Structr, Gephi, Graffeine, etc. In contrast, the VRiDaM approach is more generalized to work across heterogeneous DB types, thus permitting users to only ramp up on one tool, it is cross-platform and provides an immersive web-based VR experience. Furthermore, in contrast to the 3D DIVA or 2D Schemaball, our approach avoids the visual connection 'yarnballs' as the number of connections and tables scale.

Work on big data visualization techniques in conjunction with VR is discussed by Olshannikova et al. in [23]. Herman, Melançon, and Marshall [24] survey work on graph visualization and navigation for information visualization. In contrast, our focus is displaying the database-model structure, not on displaying and analyzing large amounts of data per se.

III. BACKGROUND INFORMATION

A. Benediktine Space

Benediktine space transforms or maps an information object's attributes to extrinsic (e.g., Cartesian coordinates, time) and intrinsic (e.g., size, shape, color) information spatial dimensions. To keep objects from overlapping, mapping principles include exclusion, maximal exclusion, scale, and transit [7]-[10].

B. WebVR

WebVR is a Mozilla JavaScript API that enables web browsers to access VR hardware. A-Frame is an open source framework for displaying VR content within HTML. It is based on an entity component system architecture in which each object in a scene is an entity (a container) consisting of components that provide desired functionality or behavior for that entity. A-Frame uses the three.js library to provide 3D graphics display in the browser and simplify WebGL programming. Systems are data containers. In contrast to game or PC station VR solutions, the use of VR within web browsers is relatively new, thus in deciding on the visualization techniques to use we considered the limitations and available capabilities and performance offered with WebVR for standard hardware (such as notebooks) that developers might use. Visualization considerations included selecting what and how many objects are displayed at any given time. Furthermore, in contrast to games, there is no real upper limit on the number of simultaneous entities a database or database model may have, which may overtax the computing and rendering capabilities and have negative impacts on the frame rates in VR, making the experience unsatisfactory and possibly resulting in VR sickness.

To characterize WebVR performance, we experimented with the implementation, some measurements of which are shown in Table I. They are averaged across 10 measurements for 10 tables with 50 columns each on a notebook with Intel Core i5-3320M 2.6Ghz, 8GB RAM, Win7 x64, Intel HD Graphics 4000, Chrome 60.0.3112.113 and A-Frame 0.6.1.

TABLE I.   AVERAGE A-FRAME PERFORMANCE (FRAMES PER SECOND)

Variants	Loading (fps)	Running (fps)
spheres, no edges	25	61
spheres, with edges	21	53
labeled spheres, no edges	11	19
circles, no edges	25	57
spheres, no edges (10x nodes)	3	12

Based on our experience and measurements with the A-Frame implementation, the following performance findings were made and affected our solution: 1) the number of rendered objects has a major impact on performance, 2) edges have a negative effect on performance, 3) text labels have a severe impact, 4) circles and spheres are equivalent.

For Finding 1, that implies that only the minimum number of objects should be displayed. Thus, rows (data values) will only be shown for selected column, not for all columns. Due to Finding 2, objects will be displayed without edges and connectors between objects will be avoided (force-graph). Due to Finding 3, text will be limited and the data value only shown when a circle (tuple) is selected.

IV.   SOLUTION APPROACH

Visualization, especially VR with its wide viewing angles, can leverage peripheral vision for information, whereby visual data is both consciously and unconsciously seen and processed. If leveraged well, visualization can be cognitively easier in providing insights than an equivalent textual analysis would require. Traditional text-centric tabular formats or boxes in E-R diagrams do not explicitly take advantage of such visual capabilities. Also, if the contents of each database table were visualized as a rectangular 2D object, as it scales both in number of tables and the size of various tables, lucidity issues occur that nullify the advantage of VR visualization.

To provide some background context for our VRiDaM approach, we describe several perspectives that were considered. Information can be grouped and large amounts of information provided in a relatively small amount of graphical space. Yet humans are limited in their sensory perception and focus, and thus visualization considerations include: determining the proper balance for what to place into visual focus in which context, what to place into the periphery, what to hide or show, and to what extent and at what point should what be visualized. To develop actionable insights from visualization, the knowledge crystallization cognitive process involves acquiring information, making sense of it, creating something new, and acting on it [26]. Regarding visual perception, gestalt psychology [27] is based on the four principles of emergence, reification, multistable perception, and invariance. Formulated gestalt grouping laws regarding visual perception include proximity, similarity, closure, symmetry, common fate, continuity, good gestalt, past experience, common region, and element connectedness. For visual representation, we considered Don Norman's design principles, in particular affordance, consistency, and mapping [28]. Other concepts considered were [10] information space,

cognitive space, spatialization, ordination, and pre-attentive processing, which refers to the accumulation of information from the environment subconsciously [29]. Visualization techniques explicitly analyzed with regard to their appropriateness for displaying data models in VR included books, cone trees, fisheye views, information cubes, perspective walls, spheres, surface plots/cityscapes/3D bar graphs, viewpoints, workspace/information space/3D rooms, worlds in worlds, and Benediktine space.

Our VRiDaM solution approach is shown in Fig. 1 and involves transforming the raw data and its metadata to internal structures (the first two being purely data forms), and then mapping these to visual element structures, and transforming these to the views seen be the user (the last two being visual forms). To adjust the views, the user provides interaction input affecting one of the aforementioned transformation steps.
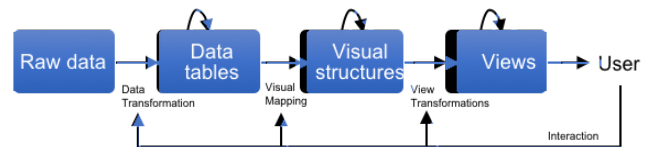
Figure 1.   VRiDaM architecture based on the information visualization reference model by Card et al. [30].

Our VR approach works across different databases products and database types (SQL and NoSQL), thus familiarity with a single VR app could be leveraged across the various database types. Alternatively, currently developing unique VR app tools for each database and/or database type would be exorbitant relative to the number of software engineers that have VR capabilities and have database-model interests, and inefficient from a learning/training perspective.

V.   SOLUTION CONCEPTS

Native and browser-based VR provide different capabilities or degrees thereof, and thus have a different potential in what can be done and what constraints and issues arise. Typically, a native VR setting will provide a more mature platform with greater VR capabilities, better hardware integration, and better performance optimization. The platform is optimized for VR, whereas a WebVR solution tries to add in VR into a browser-focused experience that was not originally designed to support VR. Thus, each of our solution concepts explores and applies different solution principles to make the most of the potential within the constraints of the environment.

In both concepts, cubes are used to represent database tables (collections for document stores, or labels for graphs), analogous to cube furniture that can be used as a table.

A. VRiDaM-N Concept

For the VRiDaM-N solution concept, spatial proximity between cubes was not used to indicate closer relations, rather an initial circular placement was used and lines/pipes used to indicate which are related. The following principles (P:) were applied:

Every table is rendered as a generic cube with a rotating label on top. This permits the use of the cube sides for

graphical icons or pictures to support a more visual experience from various angles, yet the name can usually be well read from the various angles.

Customizable icons placed on the cube sides can be used to indicate the kind of data held in a table. These can provide a custom graphical indicator as to the kind of data the table holds, without needing to read the table label. Thus, users can more rapidly identify tables. From the rear, the icons are not seen, rather attribute (column) names, but the table labels can be read. A default icon (empty table) can be used when no other is specified.

Attributes (columns, or keys for document or graph stores) are indicated as names are placed spatially behind the table cubes as an attached cube. In case the attribute names are of interest, the user can make these visible and navigate to the sides or rear of the table cube to read the attribute names, with a longer list resulting in a deeper cube, providing a visual cue that the attribute set is more extensive. When viewed from the front, the attribute names are not visible, thus simplicity is provided based on perspective when desired. Because the native rendering of text is better and more readable from various angles, we chose to stack the attribute labels on the sides and rear and keep them visible, rather than spatially separating them as a network as we did in the WebVR concept. An indicator (red colored attribute name text) can be used as a differentiator to indicate foreign keys, and similarly indexes could be shown with a different color.

Relations are visualized as a network. Pipes (equivalent to 3 dimensional lines) are used to indicate foreign key relations.

Custom grouping. Logically-related tables can be grouped by the user as desired within colored label frames.

Virtual tablet for details and accessing data tuples (rows). To view table details or raw data, the data content (tuples or rows) can be viewed via a horizontally expandable virtual tablet that can show all attributes, while supporting vertical scrolling to view any data the table holds. We decided against showing all data on a plane, since deeper text cannot be read anyway and the scrollbar can indicate how much data exists.

B. VRiDaM-WebVR Concept

The WebVR solution concept utilizes a browser-based WebVR implementation to enable cross-platform access to VR content. It assumes the user has a VR headset, but not necessarily VR controllers. Software engineers often work across different operating systems (Windows, Linux, etc.), and this permits them to utilize the app from any platform with appropriate WebVR browser support. With the WebVR solution, we struggled with readability and adequate performance when much text was rendered, and navigation and flythrough was not as smooth, and thus chose utilizing other visual components to convey information and to reduce the amount of visible text.

For the WebVR solution concept, we applied the following principles (P:):

Leverage spatial visualization in VR using a Benediktine spatial object placement approach. Our approach leverages the additional dimensional visualization and navigational capabilities VR provides (within current limitations of WebVR). Specifically, we utilize a Benediktine space visualization technique [7] with visual object spatial placement based on extrinsic spatial dimensions, whereas other entity properties are represented by intrinsic dimensions (form, size, color, etc.). The principle of exclusion ensures no two objects occupy the same spatial location, and the principle of maximal exclusion ensures that different data items are separated as much as possible [8].

Leverage dynamic self-organizing force-directed graph visualization to indicate the strength of relationship between objects via proximity. For visualizing relations, dynamic self-organizing force-directed graph placement [25] can be used in place of connectors to indicate via proximity more strongly related entities from those that are less- or unrelated. This is combined with visual highlighting of related objects when selecting an object. In this way we intend to avoid the "ball of yarn" issue with visual connectors as database models scale, or that a circular placement of many tables is no longer perceivable or meaningful to the user.

VI. IMPLEMENTATION

For the following prototype implementation screenshots, the Northwind Trading sample database consisting of 13 tables and 6635 records was used for testing and visualizing data. Additional features that can be added to our prototype in future work includes support for visualizing constraints and indexes.

A. VRiDaM-N Concept

To implement the native VR solution concept, the Unity game engine with the SteamVR plugin and runtime was chosen due to its multi-platform support, direct VR integration, popularity, and cost. For testing with VR hardware, we used the HTC Vive, a room scale VR set with a head-mounted display and two wireless handheld controllers tracked by two base stations.

Fig. 3 shows the default circular placement after loading with the front view. Cubes in purple are labeled on top with a rotating label that aligns to the camera position. The cubes can have a custom icon (such as us_states which shows the outline of the continental USA), or a default empty table icon such as categories. Table relations are shown via the connecting black pipes. The black cubes showing attribute names can be turned off if desired. Because we wanted to emphasize the icons and increase contrast, we chose a white space as the background rather than black for instance.

Fig. 4 shows a partial side perspective. The black cubes behind the purple table cubes represent attributes and contain the attribute names, with the higher black cubes representing a larger set of attributes (e.g., see employees, orders, or customers vs. the smaller shippers table). Thus, one has a visual indication of which tables have fewer attributes.

Fig. 5 shows the view from the rear, showing attribute names and any foreign key relations via red text and pipes drawn to the related key.

In Fig. 6, a virtual tablet is shown, that shows the data contents and details of the selected table. This tablet can be placed where spatially desired, and can expand when needed to show all attributes of a given table.

Fig. 7 shows the grouping capability, where the color and label of a frame can be chosen, and the cubes representing tables can be moved and logically organized as desired by the user.

Fig. 8 shows a side view where attributes can be seen within the grouping frames.

The solution concept is generic; SQL support in the native VR prototype was implemented, but due to resource and schedule constraints support for NoSQL databases will be implemented in future work.

B. VRiDaM-WebVR Concept

To implement the WebVR-based prototype, A-Frame and D3.js were utilized, which produces dynamic, interactive data visualizations in web browsers. For a self-organizing force-directed graph, our implementation uses the d3-force-3d physics engine from D3. Firefox and Chrome were used as web browsers. For database connectivity, the Spring Framework 4.3.1 was used and tested with PostgreSQL, MSSQL, MongoDB, Cassandra, and Neo4j. Content for the force-directed graph component was transformed to JSON format. Fig. 9 shows the class diagram regarding database integration.

The following visual object forms were selected:

Cubes are used to represent database tables (collections for document stores, or labels for graphs), analogous to cube furniture that can be used as a table (Fig. 10).

Cylinders are used to represent database attributes (columns) (set of similarly typed data, known as keys for document stores or graphs), analogous to columns in a building (Fig. 11).

Planes are used for projecting the database data records (rows, tuples, or entries - the actual data values) since these can be very large in both columns and records and would thus occupy a large amount of VR space as seen in Fig. 12). A plane supports maximum readability and permits VR navigation around it.

As to navigation and interaction, users can move objects as desired using standard VR controllers (we used an HTC Vive) or can use a mouse and keyboard. As seen in Fig. 13, besides using spatial proximity to indicate closer associations, if a user selects an object, that object and all its directly referenced objects are highlighted. A key image is provided as an affordance and, if selected, a popup shows the primary and foreign keys names. If desired, lines can optionally be used to emphasize relations as shown in Fig. 14.

The configuration menu is overlaid and can be used to connect to a database and query (e.g., SQL, Cypher, etc.) by typing on the keyboard and executed via enter. In order to quickly find a table, they are listed on the menu for selection and navigation to the visual object.

Fig. 15 shows the VRiDaM visualization for MongoDB document store with dbkoda example data [26] (a Northwind port was no longer available), while Fig. 16 shows Neo4j graph database with Northwind example data.

VII. EVALUATION

To evaluate VRiDaM, we compared its usage with a typical 2D database tool, DbVisualizer 10.0.13 (Free). The Northwind database was loaded to provide an impression of the sample model's complexity – the figures are not meant to be readable. Fig. 17 shows the hierarchic view, Fig. 18 the circular view, Fig. 19 the circular view with table names only, Fig. 20 the organic view, and Fig. 21 the orthogonal view.

A convenience sample of eleven computer science students was selected. All indicated they had some familiarity with SQL and they lacked NoSQL experience, so we chose to compare VRiDaM-WebVR with the common SQL tool DbVisualizer. Three had not experienced VR before. One experienced VR sickness symptoms and thus only the remaining ten were included in the results. The subjects were randomly selected to start in either VR mode (6) or the common tool (4). PostgreSQL Version 10 with the Northwind sample database was used. Java 8 update 151, Apache Tomcat v8.0, AFRAME 0.8.2, Firefox 61, and SteamVR Version 2018-05-24 (1527117754).

These database tasks were given verbally and equivalent but not the exact same five questions asked in the other mode:

1) Which tables have a relation to table X?
2) To which table(s) does the table X have a relation?
3) What columns does table X have?
4) What are the foreign or primary keys of table X?
5) What are the keys in table X?

TABLE II.   VRIDAM VS. DBVISUALIZER TASKS (AVERAGED)

	VRiDaM-WebVR	DbVisualizer
Task duration (mm:ss)	4:48	1:46
Cumulative answers given (total/incorrect/missing)	130/6/4	140/1/6
Task correctness	92%	95%

The tasks results are shown in Table II. VRiDaM-WebVR task duration was 2.7 times longer, and this can be expected since VR requires navigation time through space that 2D tools do not incur. The number of correct answers across the five tasks were 13 in VR and 14 in DbVisualizer, with ten subjects resulting in 130 or 140 cumulatively correct answers respectively. These longer VR task durations may be acceptable for certain user scenarios, and gives insight into what liabilities can be expected. A correctness difference of 3% across ten subjects is not necessarily significant and shows that the users were able to immerse themselves within minutes into a Benediktine space paradigm and perform tasks effectively. The task correctness differences could be attributed to personality, human alertness, distraction, or other factors beyond the paradigms or VR influence, as only 4 subjects in VR and only 3 subjects in non-VR were responsible for all errors, the rest had no mistakes.

Subjective impressions are shown in Fig. 2 for VRiDaM-WebVR intuitiveness and suitability of the controller interface and visualization as well as overall enjoyment. We note no significant difference between the interaction and the visualization intuitiveness or suitability. Only one subject preferred VRiDaM-WebVR. This may indicate that more training and experience would be needed for them to feel more comfortable with a WebVR tool than with a 2D tool. Various debriefing comments indicated that the Benediktine spatial

arrangement was either liked or not an issue for the subjects. When debriefed about what they liked about VRiDaM-WebVR, comments included that it was a better database-model visualization, that tables were real objects instead of text boxes, how tables where displayed in space, and the highlighting effect.
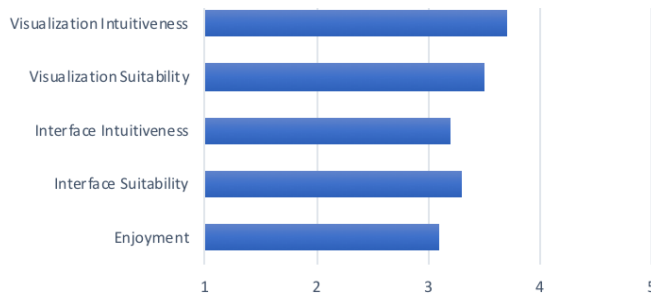
Figure 2. Average responses for VRiDaM-WebVR (scale of 1 to 5 with 5 best).

The evaluation shows some of the challenges in utilizing VR for database-model visualization and interaction. VR object interaction is not standardized, nor do users have familiarity and experience with it as they do with 2D mouse-based user interfaces. While VR enables new immersive paradigms and metaphors, these are not necessarily immediately intuitive. VR movement (moving the camera perspective) is more time consuming than scrolling or zooming a 2D perspective. For simpler tasks, VR tends to require more interaction time to accomplish the same task and thus entails efficiency costs. A 3D space permits objects to hide other objects, and for opaque objects requires movement to determine that no other objects are hidden. Given that the subjects were already familiar with E-R diagrams and SQL tools, yet had no prior training with VRiDaM-WebVR and

Benediktine space, we are satisfied with the ratings on suitability and intuitiveness.

VIII. CONCLUSION

This paper contributes VRiDaM, an immersive VR database visualization approach. Since native VR and WebVR have different capabilities and maturity levels at this time, two separate solution concepts were described and their principles elucidated. The VRiDaM-N, since it has better textual rendering capability and performance, can render comprehensive database information equivalent to common 2D graphical database tools, while permitting an immersive fly-through experience. Furthermore, customizable table icons provide a more visual annotation of what a table contains. For performance, the VRiDaM-WebVR solution concept seeks to reduce the required rendering overhead of objects and text, and avoids the risk of connection "yarn-balls" associated with other techniques that visualize many relations by leveraging the spatial locality of Benediktine space. The prototypes showed the feasibility of the native VR and WebVR solution concepts. The empirical evaluation of VRiDaM-WebVR showed it to be less efficient for equivalent analysis tasks while the correctness was slightly worse. Intuitiveness, suitability, and enjoyment were given a better than neutral rating on average. One case of VR sickness occurred and we hope to address it in future work.

One ongoing challenge for a generic tool approach is the plethora of non-standardized interfaces to NoSQL and other databases. However, by providing driver plugins we believe that the adaptation overhead is small in relation to the advantages of a VR visualization that VRiDaM brings. Future work includes a more comprehensive empirical study and will investigate various optimizations to improve usability, performance, and scalability.

APPENDIX
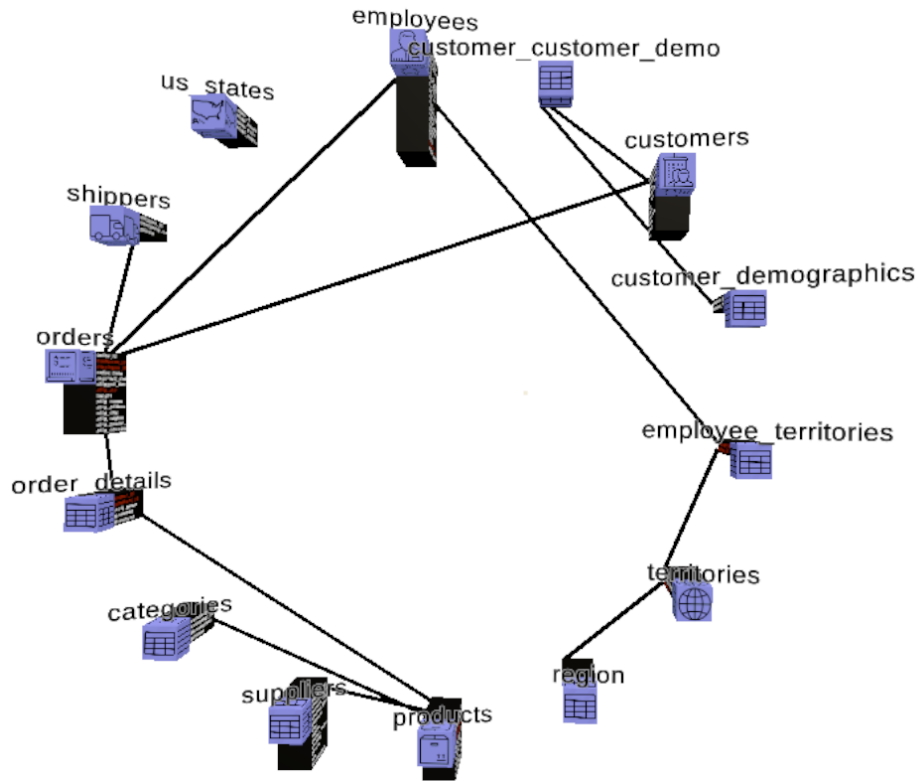
Figure 3.   VRiDaM-N front view showing Northwind tables and icons.
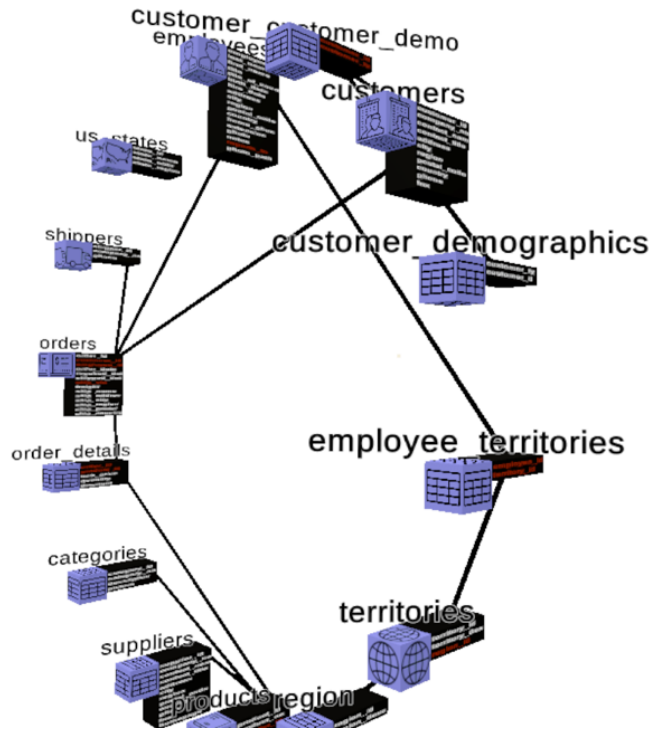
Figure 4.   VRiDaM-N side view showing Northwind tables with attributes.
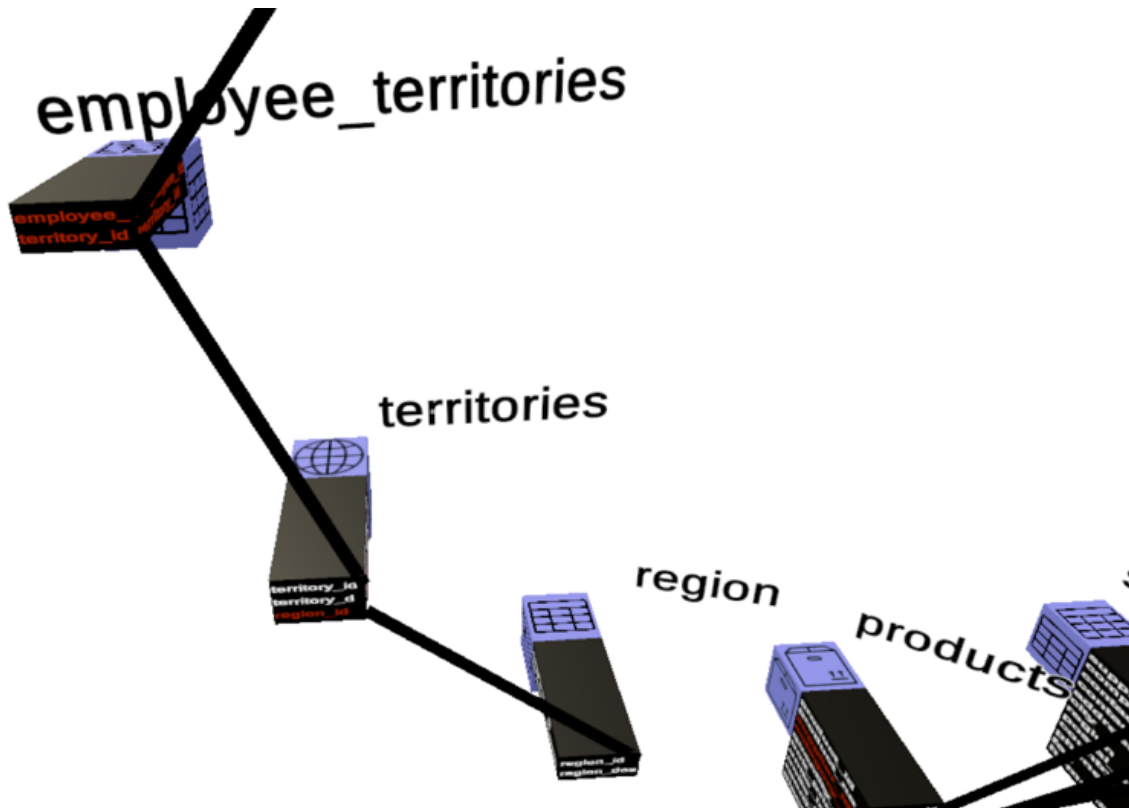
Figure 5.   VRiDaM-N rear view showing key relations.
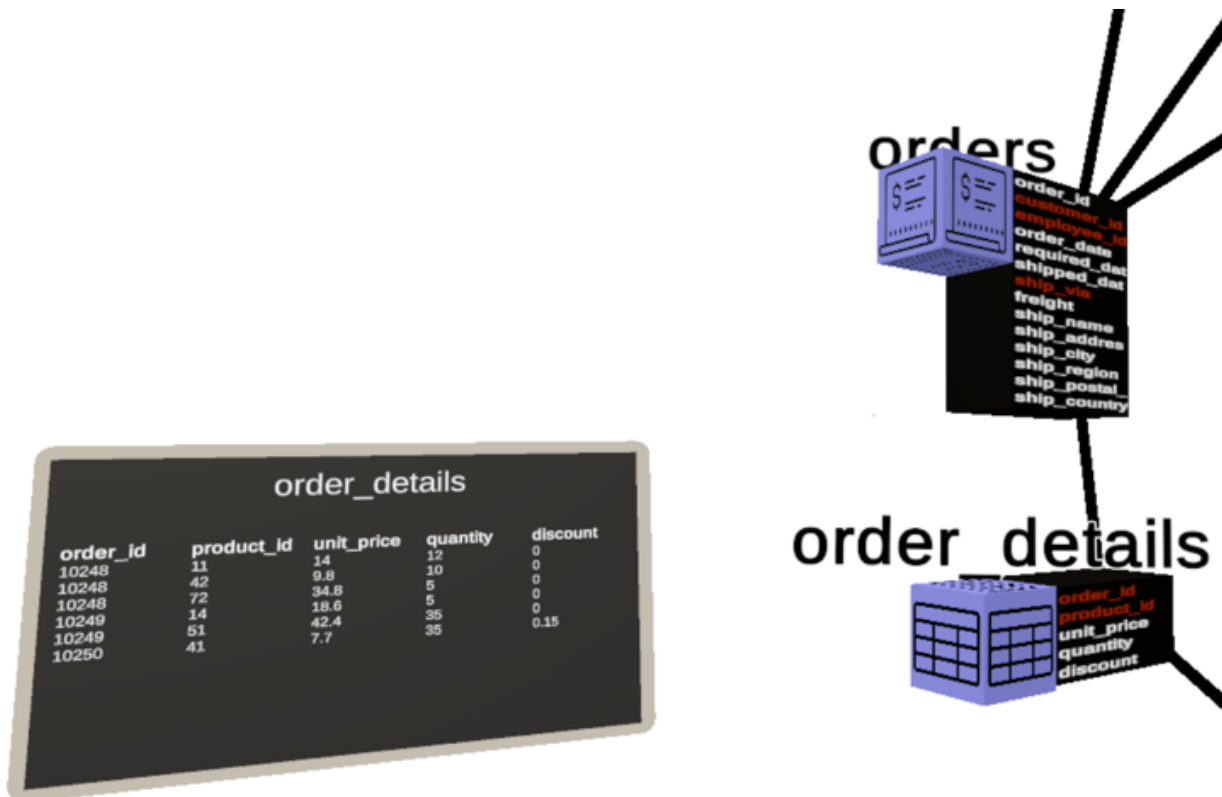
Figure 6.   VRiDaM-N with Northwind database showing virtual tablet with row/tuple data for order_details table.
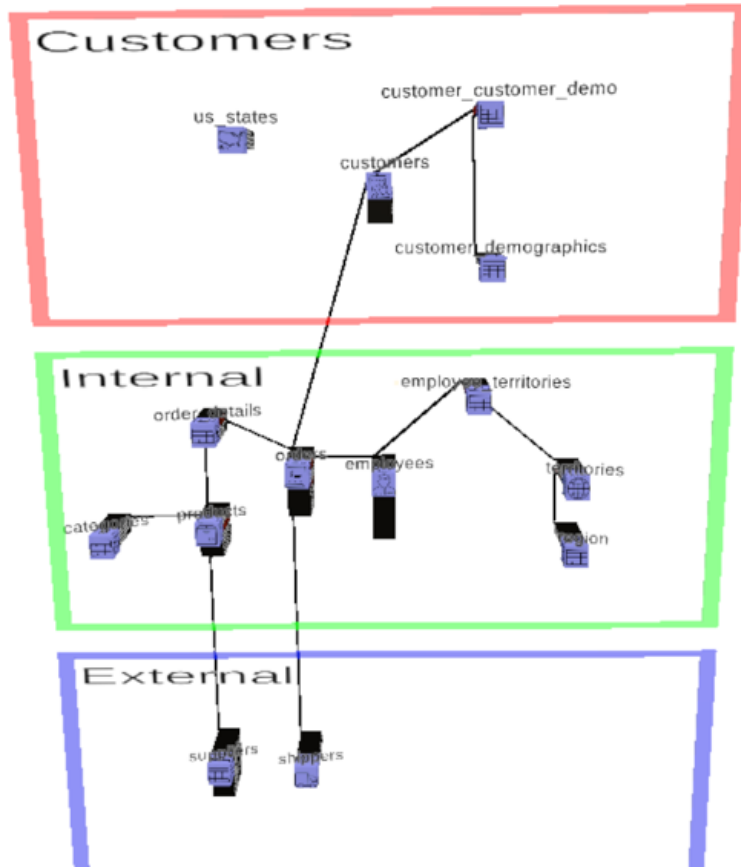
Figure 7. VRiDaM-N front view with Northwind database showing grouping and placement capability.
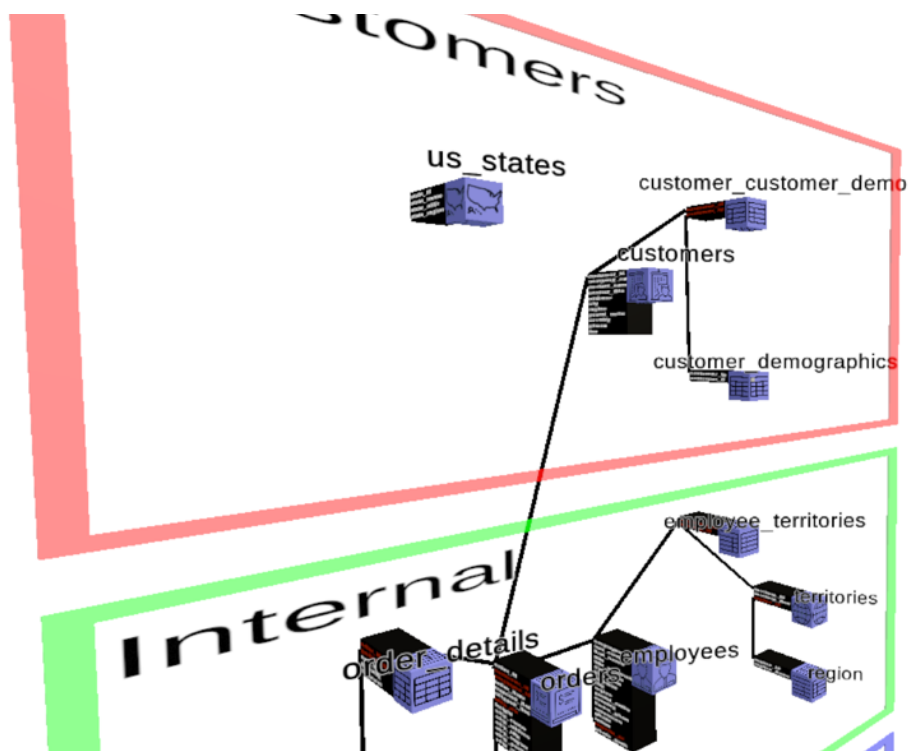
Figure 8. VRiDaM-N with Northwind database showing grouping with attribute names visible from the side.
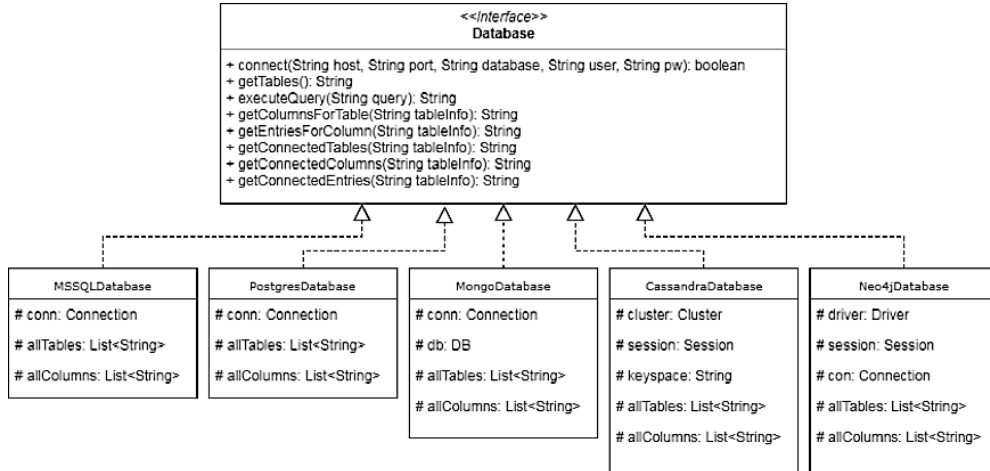
Figure 9.   Class diagram of database integration.
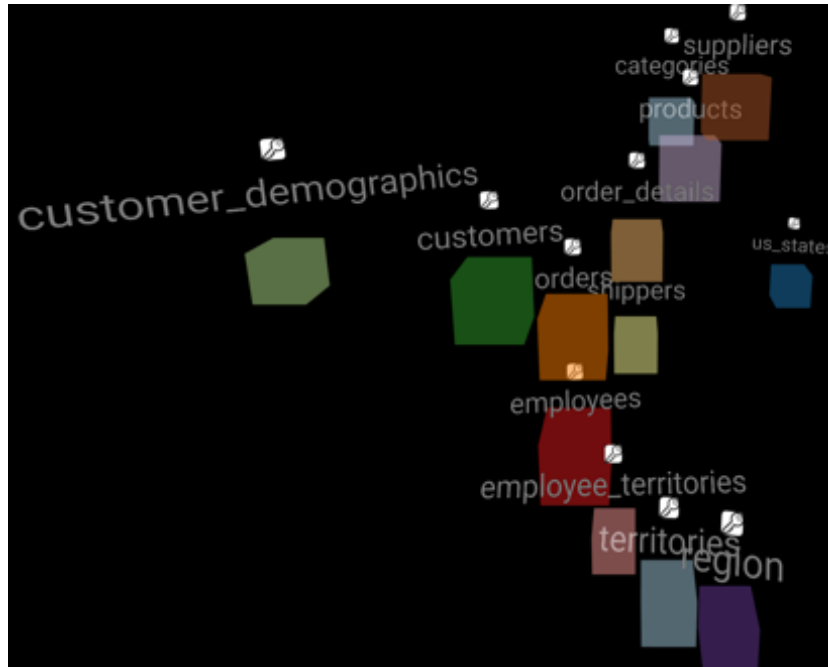
Figure 10. VRiDaM showing Northwind tables from PostgreSQL in Benediktine space.

Figure 11. Columns visible orbiting selected table (identical color).
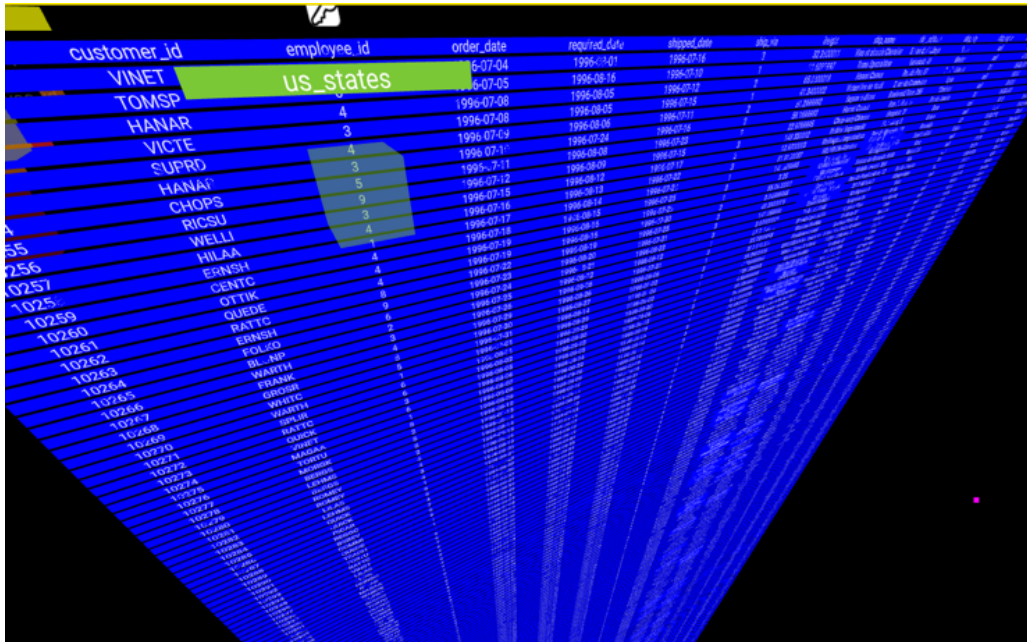
Figure 12. Table records (tuples) projected onto a plane.
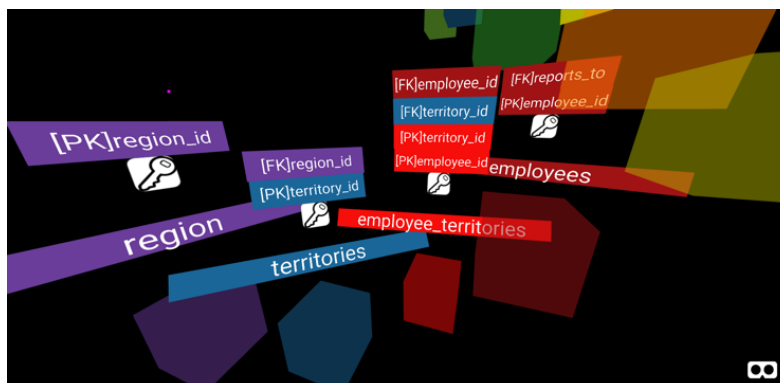
Figure 13. Primary and foreign keys for table shown as popups.
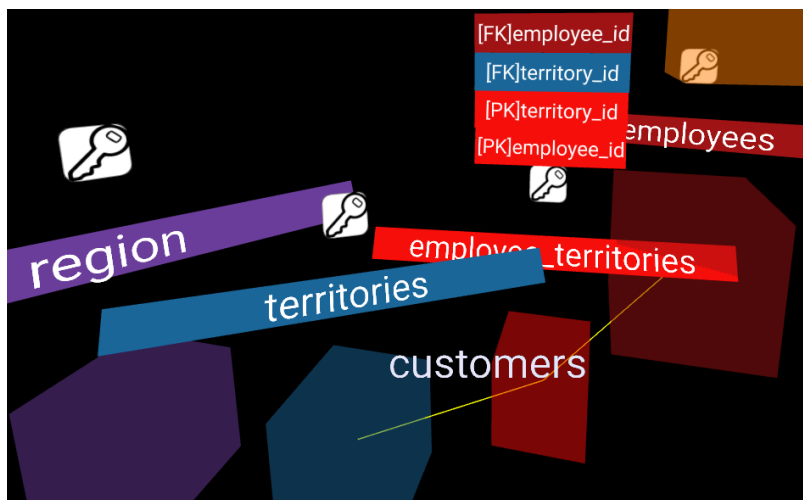
Figure 14. Example optional relation visualization using lines.
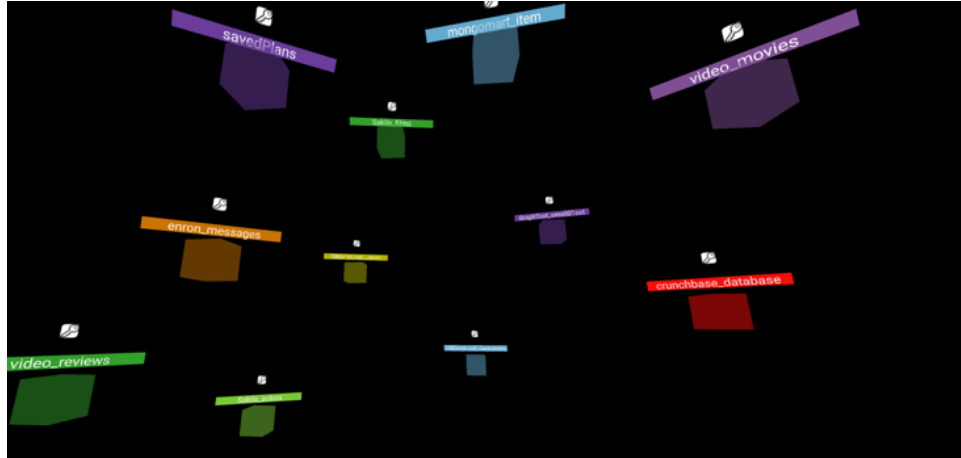
Figure 15. VRiDaM of MongoDB document store with dbkoda example data [26], showing spatial orientation and not intended to be readable.
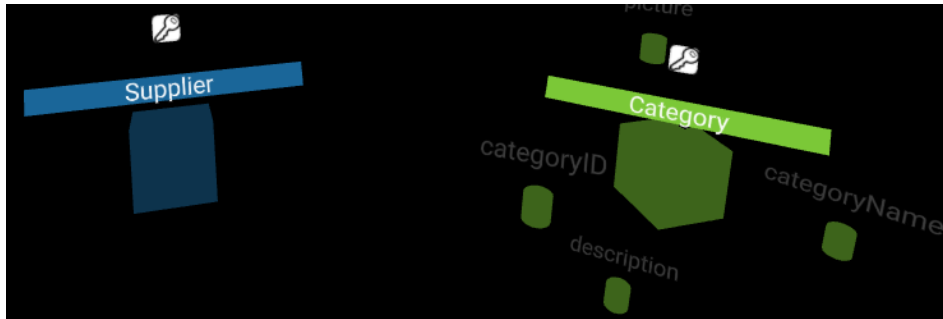
Figure 16. VRiDaM of the Neo4j graph database with Northwind example data.
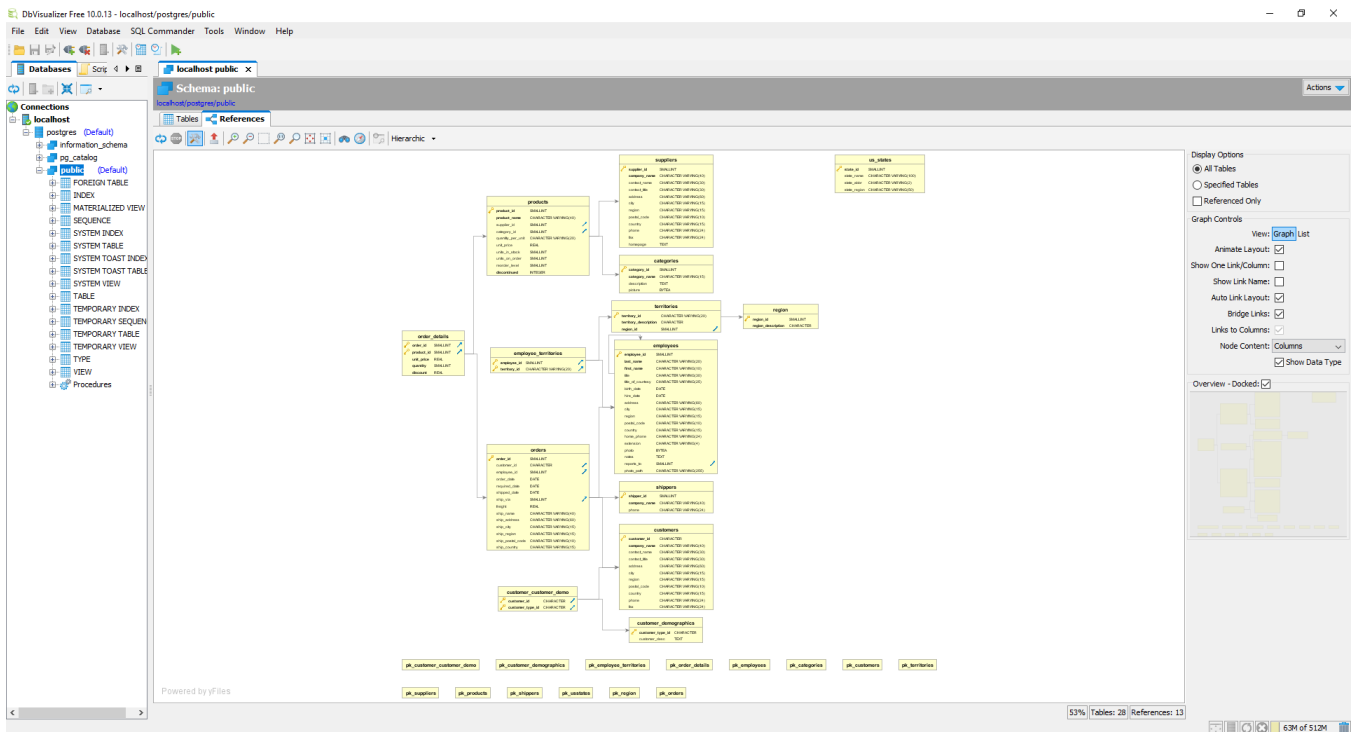
Figure 17. Northwind Traders in DbVisualizer in the hierarchic view.
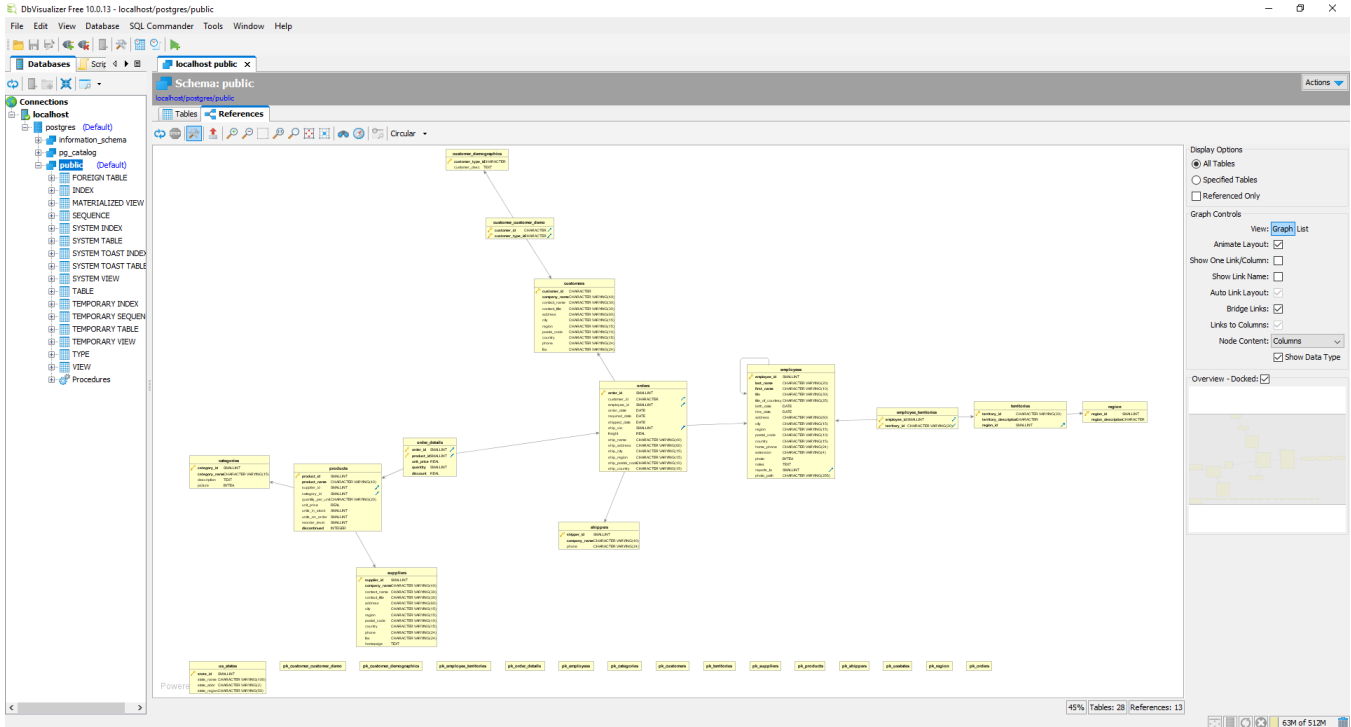
Figure 18. DbVisualizer Circular View with Northwind Traders data model.
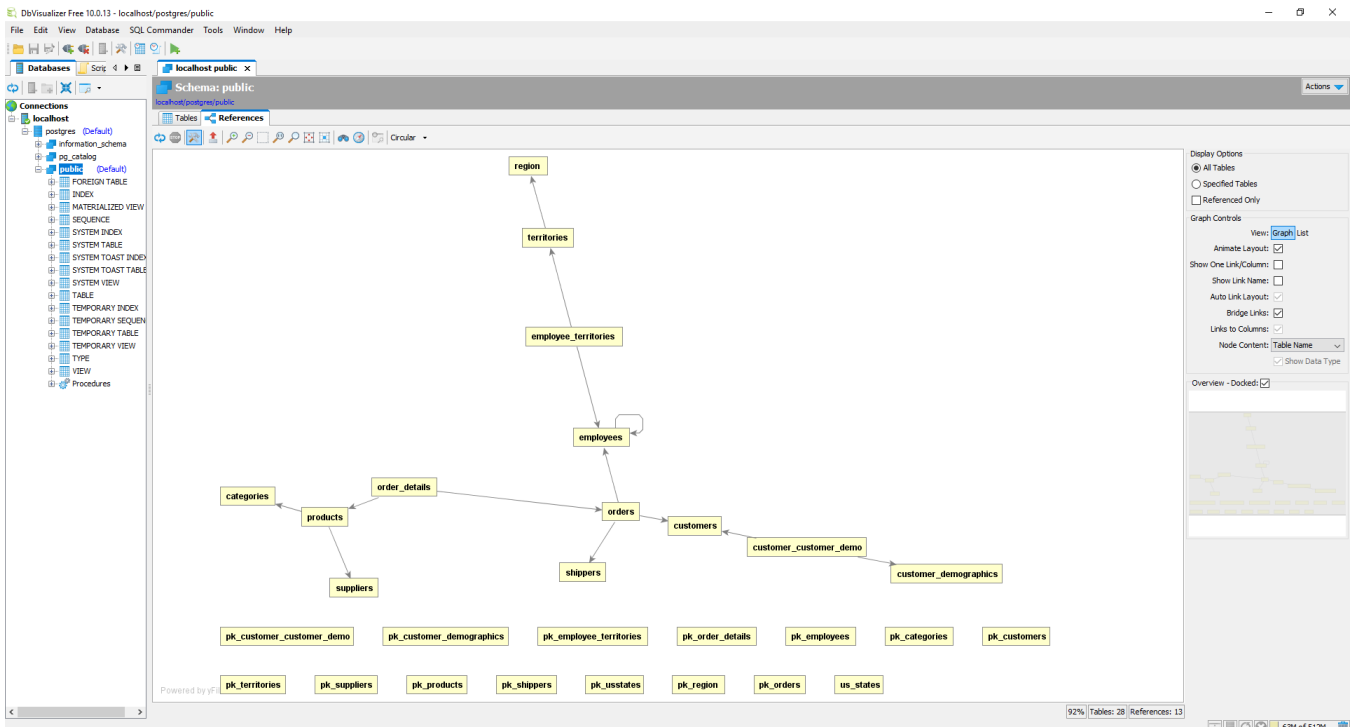
Figure 19. DbVisualizer in circular view (table names only).
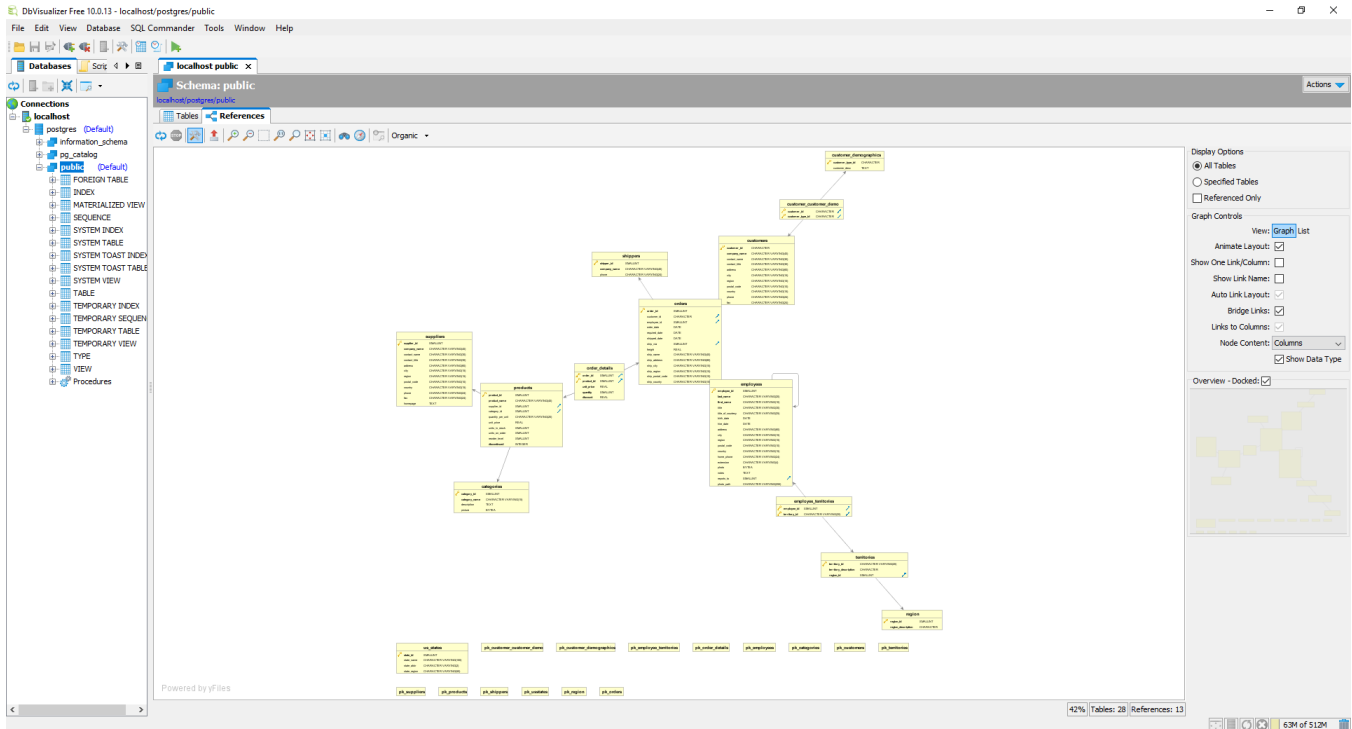
Figure 20. Northwind Traders data model in DbVisualizer in organic view.
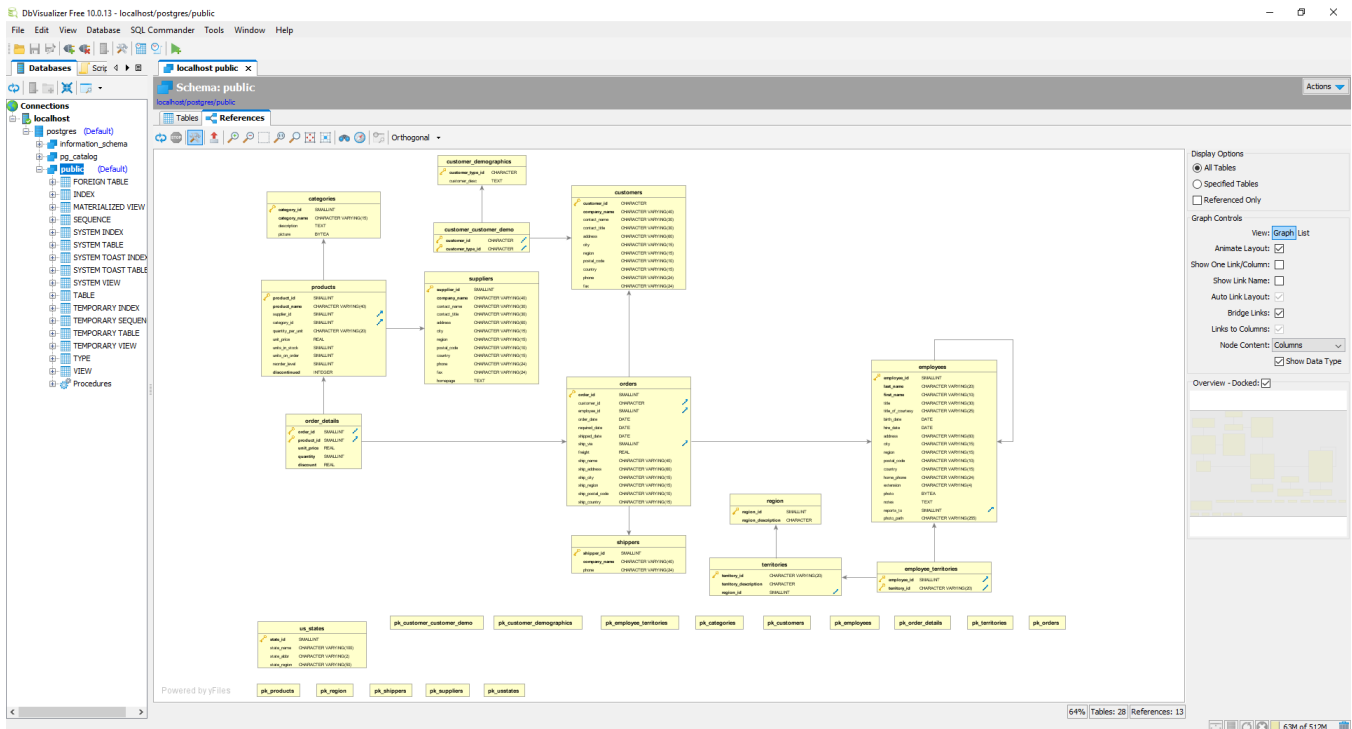
Figure 21. Northwind Traders data model in DbVisualizer in orthogonal view.

REFERENCES

[1] R. Oberhauser, "Database Model Visualization in Virtual Reality: A WebVR and Benediktine Space Approach," In: Proceedings of the Thirteenth International Conference on Software Engineering Advances (ICSEA 2018). IARIA, pp. 108-113, 2018.

[2] D. Reinsel, J. Gantz, and J. Rydning, "Data Age 2025: The Evolution of Data to Life-Critical," IDC Whitepaper (April, 2017).

[3] Cisco. The Zettabyte Era: Trends and Analysis. (Jun 7, 2017). Whitepaper. Available from https://www.cisco.com/c/en/us/solutions/collateral/service-provider/visual-networking-index-vni/vni-hyperconnectivity-wp.html 2019.12.04

[4] R. Cattell, "Scalable SQL and NoSQL data stores," Acm Sigmod Record 39, no. 4, pp. 2-27, 2011.

[5] P. Chen, "Entity-Relationship Modeling: Historical Events, Future Trends, and Lessons Learned," In Software pioneers. Springer-Verlag, pp. 296-310, 2002.

[6] T. Merel, "The reality of VR/AR growth," Jan 11, 2017. Available from http://techcrunch.com/2017/01/11/the-reality-of-vrar-growth/ 2019.12.04

[7] M. Benedikt, "Cyberspace: some proposals," In Cyberspace, Michael Benedikt (Ed.). MIT Press, Cambridge, MA, pp. 119-224, 1991.

[8] P. Young, "Three Dimensional Information Visualisation," Technical Report, University of Durham, 1991.

[9] U. Fayyad, A. Wierse, and G. Grinstein (Eds.), Information Visualization in Data Mining and Knowledge Discovery. Morgan Kaufmann, 2002. ISBN: 1558606890.

[10] S. Fabrikant and B. Buttenfield, "Formalizing Semantic Spaces for Information Access," Annals of the Association of American Geographers, 91(2), pp. 263-280, 2001.

[11] D. Butler, J. Almond, R. Bergeron, K. Brodlie, and R. Haber, "Visualization reference models," In Proc. Visualization '93 Conf., IEEE CS Press, pp. 337-342, 1993.

[12] K. Okada, M. Yoshida, T. Itoh, T. Czauderna and K. Stephens, "VR System for Spatio-Temporal Visualization of Tweet Data," In Proc. 2018 22nd International Conference Information Visualisation (IV). IEEE, pp. 91-95, 2018.

[13] K. Okada, M. Yoshida, T. Itoh, T. Czauderna and K. Stephens, "VR system for spatio-temporal visualization of tweet data and support of map exploration," In: Multimedia Tools and Applications. Springer, pp. 1-20, 2019.

[14] A. Moran, V. Gadepally, M. Hubbell, and J. Kepner, "Improving Big Data visual analytics with interactive virtual reality," In Proc. 2015 IEEE High Performance Extreme Computing Conference (HPEC). IEEE, pp. 1-6, 2015.

[15] B. Sun and W. Xu, "An Immersive Visual Analytics Platform for Multidimensional Dataset," Available from https://www.osti.gov/servlets/purl/1542284 2019.12.04

[16] A. Bierbaum et al, "VR Juggler: A virtual platform for virtual reality application development," In Proc. IEEE Virtual Reality. IEEE, pp. 89-96, 2001.

[17] F. Fittkau, A. Krause, and W. Hasselbring. "Exploring software cities in virtual reality," In Proc. IEEE 3rd Working Conf. on Software Visualization (VISSOFT). IEEE, pp. 130-134, 2015.

[18] A. Teyseyre and M. Campo, "An overview of 3D software visualization," In IEEE Trans. on Visualization and Computer Graphics. IEEE, 15(1), pp. 87-105, 2009.

[19] A. Kashcha, "Software Galaxies," Available from http://github.com/anvaka/pm/ 2019.12.04

[20] J. Rilling and S. Mudur, "On the use of metaballs to visually map source code structures and analysis results onto 3d space," In Proc. 9th Work. Conf. on Reverse Engineering. IEEE, pp. 299-308, 2002.

[21] P. McIntosh, "X3D-UML: user-centered design, implementation and evaluation of 3D UML using X3D," Ph.D. dissertation, RMIT University, 2009.

[22] M. Krzywinski, "Schemaball: A New Spin on Database Visualization," In Dr. Dobb's: The World of Software Development, 2004.

[23] E. Olshannikova, A. Ometov, Y. Koucheryavy, and T. Olsson, "Visualizing Big Data with augmented and virtual reality: challenges and research agenda," J. of Big Data, 2(22), 2015.

[24] I. Herman, G. Melancon, and M. Scott Marshall, "Graph visualization and navigation in information visualization: A survey," In IEEE Transactions on visualization and computer graphics, 6(1), pp. 24-43, 2000.

[25] G. Di Battista, P. Eades, R. Tamassia, and I. Tollis, Graph drawing: algorithms for the visualization of graphs. Prentice Hall PTR, 1998.

[26] S. Card, "Information Visualization," In: Human Computer Interaction Handbook: Fundamentals, Evolving Technologies, and Emerging Applications (Ed: Jacko, J.), Third Edition. CRC Press, pp. 544-545, 2012.

[27] S. Palmer, Vision Science. MIT Press, Cambridge (USA) 1999, ISBN 978-0262161831.

[28] D. Norman, The Design of Everyday Things: Revised and Expanded Edition. Hachette UK, 2013.

[29] A. Van der Heijden, "Perception for selection, selection for action, and action for perception," Visual Cognition, 3(4), pp. 357-361, 1996.

[30] S. Card, J. Mackinlay, and B. Schneiderman (editors), Readings in Information Visualization: Using Vision to Think. Morgan Kaufman, 1999.

[31] [Online] Available from https://github.com/SouthbankSoftware/dbkoda-data 2019.12.04

Implementing Hand Gestures for a Room-scale Virtual Reality Shopping System

Chunmeng Lu
Waseda University
Kitakyushu, Japan
Email: lcm0113@163.com

Boyang Liu
Waseda University
Kitakyushu, Japan
Email: waseda-liuboyang@moegi.waseda.jp

Jiro Tanaka
Waseda University
Kitakyushu, Japan
Email: jiro@aoni.waseda.jp

Abstract—In the virtual reality (VR) environment, the user is required to input information and achieve interaction with virtual objects. At present, most VR systems provide some input devices, such as keyboard and controller. However, utilizing such devices is not intuitive, especially in the case of a VR shopping system. In real applications, we use our hands to handle objects. In virtual applications, using hand gestures to interact with a VR shopping store will provide us a more intuitive VR shopping experience. Following the needs of the room-scale VR shopping activities, we have introduced a new gesture classification for the gesture set, which has three levels to classify hand gestures based on the characteristic of gestures. We have focused on the gestures in level 3. We have built a room-scale VR shopping system and applied the new hand gesture set for the interaction in the VR shopping system. We conducted experiments to evaluate the accuracy of gestures in a VR shopping environment. The classification and set of gestures together when evaluated showed that these specific gestures were recognized with high accuracy.

Keywords- Room-scale Virtual Reality Shopping; Gesture set; Gesture classification; Gesture interaction.

I. INTRODUCTION

Currently, people can roam in the virtual environment through Head-Mounted Display (HMD). As shopping is one of the most important activities in the real world, a virtual shopping environment can be a part of the virtual environment. We are familiar with e-commerce or online shopping. We can extend online shopping to the virtual environment [1] because the virtual reality (VR) shopping experience has the potential to allow users to surpass geography and other restrictions.

With the improvement of VR technology, many researchers and companies attempt to apply VR technology in the e-commerce field to find profitable economic value. Alibaba is a famous IT company and is known for its online shopping services. Tianmao is one of its online shopping services. Tianmao presented a VR shopping application, called Tianmao buy+ [2], for smartphones. The Tianmao buy+ attempted to combine the convenience of online shopping and the facticity of physical store shopping. From simple and cheap VR devices and smartphones, people in China can view stores around the American Times Square and pay for orders online. The VR application creates in people a feeling of the shopping experience at the American Times Square.

Some companies employ VR technology to create virtual stores. The furniture company IKEA presented a room-scale VR kitchen to show its design [3]. In the room-scale VR kitchen, a user can utilize HTC vive to view the equal proportion VR kitchen, and even interact with the VR environment compared to physical furniture stores, the VR environment can provide more functions and interactions. Users can view the kitchen freely in the comfort of their room. In the VR kitchen, a user can easily change the color of the furniture, an impossible task in a physical store.

The VR technology company, inVRsion, presents a VR supermarket system based on room-scale VR [4]. Their retail space, products, and shopping VR experience solutions provide an immersive shopping environment. In the VR shopping environment, a manager can analyze customer behavior through eye-tracking for market research insights. The system can help the seller to test his category projects, new packaging, and communication instore before implementation. A user can search his target products more easily than in a physical supermarket. This system tries to provide a method for users to view a big virtual supermarket in a room.

This article is organized as follows. In Section I, the Introduction is presented. Section II specifies the problem to be solved. Section III presents the research purposes and approaches. Section IV presents related works. Section V introduces the design of the system. Section VI introduces the implementation of the designed system. Section VII specifies the detailed process of gesture recognition. In Section VIII, we discuss how to apply gestures set in a room-scale VR shopping environment. In Section IX, we present evaluation results. In Section X, we conclude and present future works.

II. PROBLEM

The VR shopping environment provides an emulated environment in which the virtual objects are similar to physical objects. We normally use our hands to touch and grab objects around us. Thus, the use of controllers in a VR shopping environment is not sufficiently immersive (see Figure 1). The user lacks the feeling sensations to hold a virtual object when utilizing controllers.

The number and function of buttons in the controllers are limited (see Figure 2), thus restricting the interaction with products when they are used in a room-scale VR environment. Conversely, the human hand can perform various gestures for human-computer interaction. Compared to controllers, hand gestures can improve the VR immersion shopping experience and allow a greater dictionary of commands for the VR shopping system.

Figure 1. Using controllers in the virtual environment

Figure 2. The buttons of the controller

III. GOAL AND APPROACH

The main features of VR include immersion, plausibility, interaction, and fidelity. In this research, we aim to present a new hand gesture set suitable for room-scale VR shopping activity to replace the controllers and improve immersion experience. We introduce a new gesture classification functionality to achieve a more structured gesture set. We apply a room-scale shopping system to provide an immersive virtual shopping environment to simulate a physical shopping store. In the room-scale VR shopping environment, the user is able to walk in it and to view the virtual shopping environment through the HMD. In the designed new gestures for the room-scale VR shopping system, users can interact with the VR environment by natural hand gestures instead of with controllers. We introduce a gesture classification

functionality with three levels to classify hand gestures based on gesture characteristics. By summarizing the hand gestures, we obtain a new hand gesture set for room-scale VR shopping activity. The hand gesture set is expected to improve user convenience and immersion experience within a room-scale VR shopping system.

We utilize the VR devices to build the room-scale VR shopping system with two sensor stations installed in the room. These sensor stations create a walking area for the user. When moving in this area, user motion information is captured by the sensor stations. The system receives rotation and three-dimensional data coordinates from the HMD worn by the user. The sight vision in the virtual environment moves synchronously with HMD. We employ a depth sensor to recognize the hand gestures. The virtual environment allows the user to experience his virtual and physical hands moving synchronously. The system can distinguish the gestures and allow for interaction between the user and with the virtual environment.

IV. RELATED WORK

A. Virtual Reality

Virtual Reality (VR) allows for users to interact in a virtual environment as one is in the physical world. It provides an illusion of "being there" [5].

In previous years, the development of computer graphics, 3D technology, and electrical engineering, permitted improvements in HMD of VR. Recently, VR devices have gained space outside of the laboratory. Some companies have introduced simple and easy-to-use VR devices for the consumer market, such as HTC Vive and Oculus Rift.

B. VR Shopping

People can navigate in the virtual environment through HMD. In the last decades, many VR shopping environments have been presented. Some works aim at improving VR shopping experience, while others research on the interaction in virtual shopping environments. Bhatt presented a theoretical framework to attract customers through a website based on these three features: interactivity, immersion, and connectivity [6]. Chen et al. presented a VRML-Based virtual shopping mall. They analyzed the behavior of a customer in a Virtual Shopping Mall System. They also explored the application of intelligent agents in shopping guidance [7]. Lee et al. designed a virtual interactive shopping environment and analyzed if the virtual interface had positive effects [8]. Verhulst et al. presented a VR user study. They applied the VR store as a tool to determine if the user in the store wished to buy food [9]. Speicher et al. introduced a VR Shopping Experience model. Their model considered three aspects: customer satisfaction, task performance, and user preference [10].

The previous researches presented good features of VR shopping that raised interests from retail and online shopping companies for VR shopping. The company IKEA presented a room-scale VR environment in which the user can view a virtual kitchen and interact with the furniture [3]. In another

example, inVRsion provides a virtual supermarket shopping system, the Shelfzone VR [4]. In the future, more applications are expected for VR shopping.

C. Room-scale VR and Hand Gesture

When the user is moving his gaze in the virtual world with HMD, he cannot move his physical body in the real world. Thus, the absence of sensory movement experience between the virtual and physical environments reduces immersion experience in VR. This absence also causes motion sickness in some users [10][11]. Nevertheless, if walking is synchronized both in virtual and physical environments, this provides an improved sensory experience.

Room-scale VR shopping environment provides an emulated environment in which virtual objects are similar to real objects. Commonly, we use our hands to touch and hold objects around us. Thus, the use of controllers in VR shopping environment causes it not to be sufficiently immersive. The user lacks the sensation of holding virtual objects similar to physical ones when using controllers. Additionally, the number and function of buttons in the controllers are limited, which restricts the interaction when utilizing them in a room-scale VR environment. There are three buttons on a controller. To achieve VR shopping activities, these three buttons are designed for the interaction method in the VR shopping system. Compared with controllers, the use of hand gestures can improve the VR immersion shopping experience and provide a rich interaction dictionary of commands for the VR shopping systems.

Hand gestures have widely been utilized in human-computer interfaces. Gesture-based interaction provides a natural intuitive communication between people and devices. People use 2D multi-touch gesture to interact with devices such as smartphones and computers in daily life. The 3D hand gesture can be employed for some devices equipped with a camera or depth sensor. The most important point in hand gesture interaction is how to provide computers the ability to recognize commands from hand gestures [12]. Wachs et al. summarized the requirements of hand-gesture interfaces and the challenges when applying hand gestures in different applications [13]. Yves et al. presented a framework for 3D visualization and manipulation in an immersive space. Their work can be used in AR and VR systems [14]. Karam et al. employed a depth camera and presented a two-hand interactive menu to improve efficiency [15]. These previous studies show that there is potential for hand gestures in the human-computer interaction field.

In a previous work, we extended 2D multi-touch interaction to 3D space and introduced a universal multi-touch gesture for 3D space [16]. We called these midair gestures in 3D as 3D multi-touch-like gestures (see Figure 3).

To recognize 3D multi-touch-like gestures, we present a method using machine learning. However, the use of machine learning alone is not sufficient to perform the task accordingly. Although machine learning techniques can recognize the hand shape, it fails to address the begin and end timing of gestures movement. If we cannot precisely recognize the gesture timeframe, it is difficult to provide a

fast response to the performed gesture. Therefore, proper timing of events is required. We use a depth camera to detect the state of fingers to discover the begin and end timeframe of the gestures.
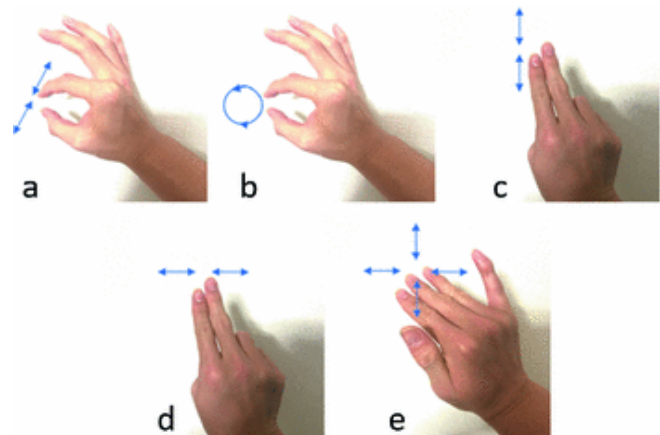
Figure 3. Five 3D multi-touch-like gestures: (a) zoom in/out, (b) rotate, (c) scroll, (d) swipe, and (e) drag

The previous related works elucidated the broad application prospects for room-scale VR shopping and gestures. The proposed work on gesture set design for room-scale VR can be used in these systems to provide immersive VR shopping experience.

V. SYSTEM DESIGN

A. Room-scale VR Shopping Environment

To achieve VR shopping activity, we design a VR shopping store as the shopping environment, similar to physical stores. We arrange desks, shelves, and goods in the VR shopping store as shown in Figure 4.

Figure 4. A VR store

The room is provided with an empty area in a real room, which is included in a 3D space. We use a tracking sensor to capture the motion and rotation of HMD in the 3D space for the VR shopping system use. As shown in Figure 5, the

length, width, and height of the 3D space is 4 m, 3 m, and 2 m, respectively. The 3D space contains walking area of the room.
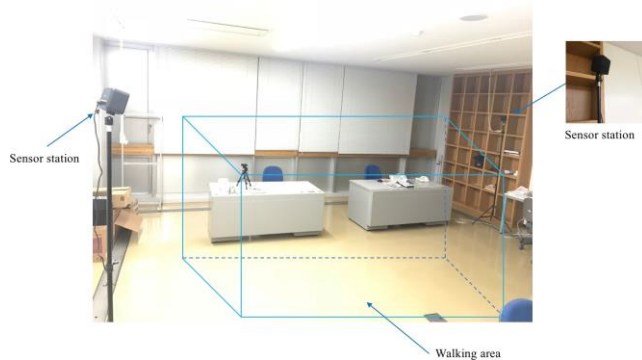
Figure 5. 3D space and walking area

In the room-scale VR shopping environment, there is also a virtual walking area, as shown in Figure 6. The virtual walking area is the same as the area in the actual room. Because the VR shopping store is larger than the real room, the user can change the virtual walking area while looking at the entire VR shopping store.
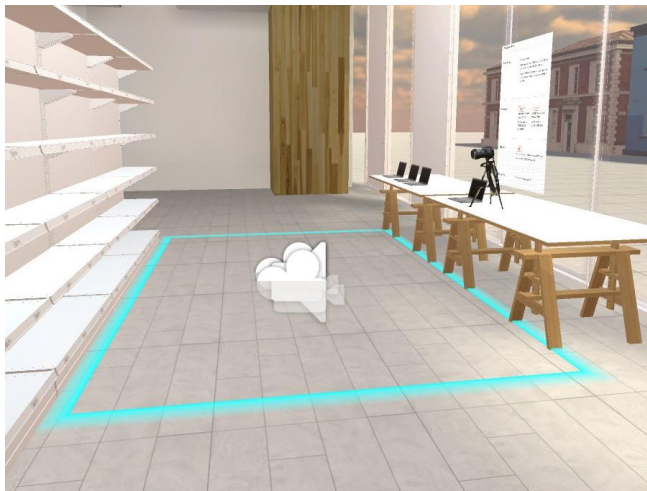
Figure 6. Walking area in VR environment

B. Gesture Set

In our research, we designed a series of hand gestures specially for the room-scale VR shopping activity. The hand gestures must provide natural and suitable interaction between the user and system. Based on the specific activities in room-scale VR shopping system, we design these 14 gestures in the system according to previous work [16]:

(1) OK gesture;
(2) No gesture;
(3) push/pull gesture;
(4) waving gesture;
(5) pointing gesture;
(6) grab gesture;
(7) holding gesture;
(8) drag gesture;
(9) rotation gesture;
(10) zoom in/out gesture;
(11) click gesture;
(12) two-fingers scroll/swipe gesture;
(13) opening/closing gesture; and
(14) changing area gesture.

These gestures create a new gesture set for a room-level VR shopping system.

C. Gesture Classification to define levels

We propose a new type of gesture classification that classifies the hand gestures according to different characteristics of gestures. There are three levels of gesture classification:

Level 1: Core static hand positions are classified into level 1. In level 1, gestures englobe hand shape without hand movements. The classic example is pointing gestures.

Level 2: Dynamic palm motions are classified into level 2. Level 2 only considers the palm movement regardless of finger shape. The classic example is pulling and pushing.

Level 3: Combined hand gestures are classified into level 3; they combine the features of level 1 and level 2 gestures. Level 3 considers the finger movement and shape and palm movement.

In the VR shopping environment, the hand gestures are divided into different levels. In Figures 8 and 9, the red arrow indicates the finger movement direction.

The gesture classification method classifies the gesture set in a structured way and provides a structure that can be used in other gesture sets in different VR systems. Based on the system structure, researchers can design appropriate gestures for their own VR systems.

1) Level 1 Gestures:

In our system, we use level 1 gestures to send feedback to the system. These gestures are static signals of the shopping system, and the system only needs to detect the hand shape. Table I shows the functions of level 1 gestures, and Figure 7 depicts two level 1 gestures.

TABLE I.   FUNCTION OF GESTURES IN LEVEL 1

Level	Gesture	Function
1	OK	Inform a positive feedback to the system.
1	NO	Inform a negative feedback to system.

(a) OK gesture   (b) NO gesture
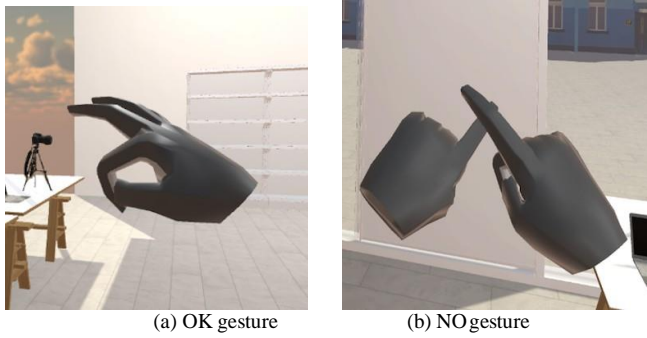
Figure 7. OK and NO gestures positive or negative feedback to system

2) Level 2 Gestures:

Level 2 Gestures are palm movements. After selecting a virtual object, the user can control or interact with level 2 gestures. Table II shows the functions of level 2 gestures, and Figure 8 shows two level 2 gestures.

TABLE II.   FUNCTION OF GESTURES IN LEVEL 2

Level	Gesture	Function
2	Push/pull	Push or pull a virtual object with a hand.
2	Waving	Make virtual object return to the original position.

(a-1) Push gesture   (a-2) Pull gesture
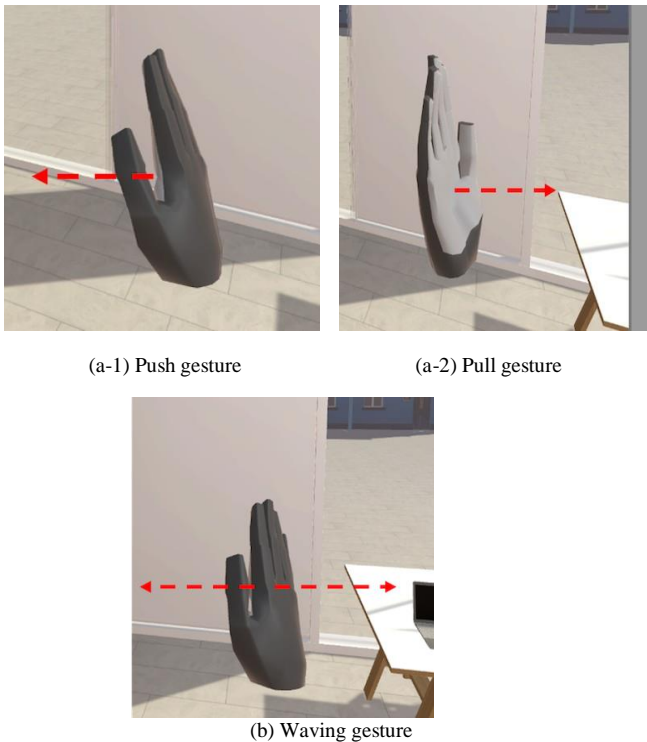
(b) Waving gesture

Figure 8. Level 2 gestures: push/pull and waving

3) Level 3 Gestures:

Gestures in level 3 combine hand gestures of finger shapes and hand movements. They combine the features of Level 1 and Level 2 gestures.

The design of a suitable and convenient gesture set for the user determines an immersive VR shopping experience. In levels 1 and 2, gestures are simple. Because level 3 gestures are more complex, they are the focus of this research.

In level 3, the system identifies hand shapes and detects finger and hands movement simultaneously.

As the gestures indicate different messages they are divided into three classification categories:

• The core gesture: pointing gesture

• Gestures for interacting with a virtual object: (1) grab gesture, (2) holding gesture, (3) drag gesture, (4) rotation gesture, and (5) zoom in/out gesture;

• Gestures for interacting with menu: (1) click gesture, (2) scroll/swipe gesture, and (3) opening/closing gesture;

• Gesture for interacting with space: change area gesture

In the gesture set, the pointing gesture is the most important gesture, because it is used to select a target object or button before any interaction, as shown in Table III and Figure 9.

TABLE III.   FUNCTION OF POINTING GESTURE

Level	Gesture	Function
3	Pointing	Point a virtual object with the index finger.
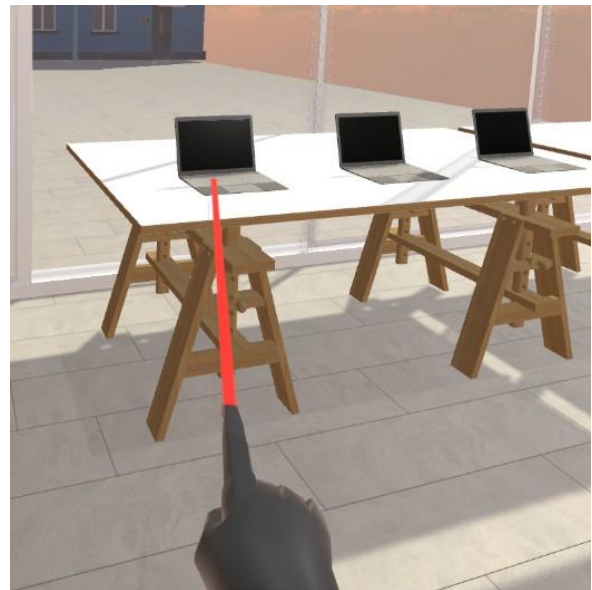
Figure 9. Pointing gesture

Some gestures are mainly used to interact with virtual objects in a VR shopping store, such as moving a virtual object. We design these gestures to achieve the following simple functions: grab, hold, drag, rotation, and zoom in/out, as shown in Table IV and Figure 10.

TABLE IV.   FUNCTIONS OF GESTURES FOR INTERACTING WITH OBJECT

Level	Gesture	Function
3	Grab	The object is approximated to the hand that grabs it.
3	Hold	Holds a virtual object with one hand.
3	Drag	Move virtual object freely with a hand.
3	Rotation	Rotate a virtual object when observing it with a hand.
3	Zoom in/out	Zooms in or out of a virtual object using the relative motion of thumb and index finger.

(a) Grab   (b) Holding

(c) Drag   (d) Rotation

(e-1) Zoom in   (e-2) Zoom out
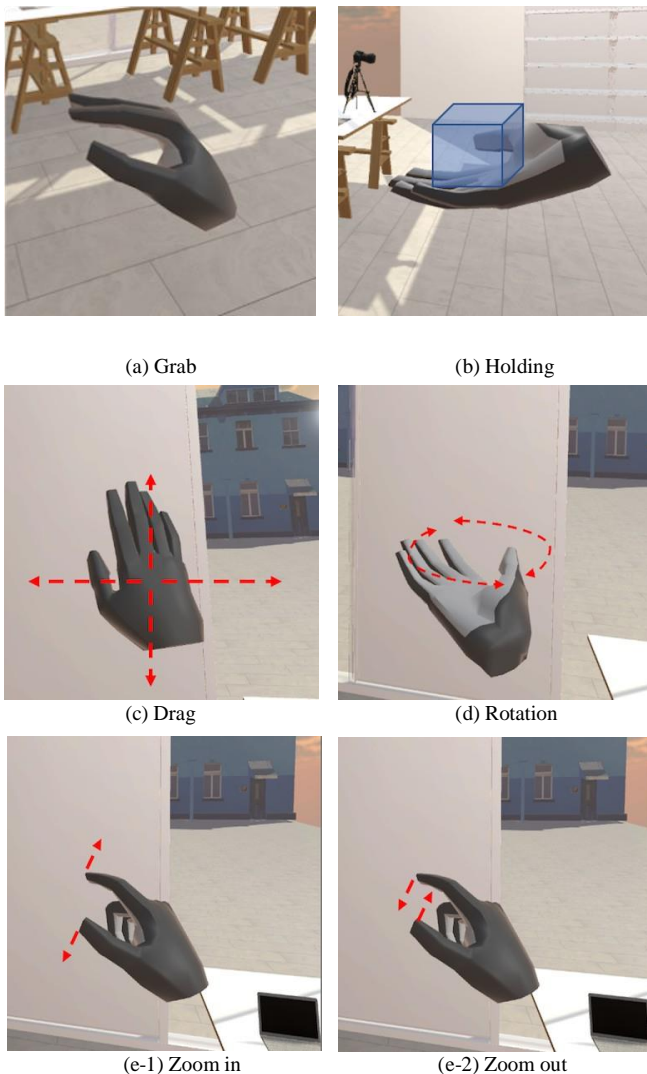
Figure 10. Gestures for interacting with object

In some cases, the user requires to interact with the menu to perform shopping activities. To achieve that, we design the following gestures: click, scroll/swipe, opening/closing, as shown in Table V and Figure 11.

TABLE V.   FUNCTIONS OF THE GESTURES FOR INTERACTING WITH MENU

Level	Gesture	Function
3	Click	Click the buttons with index finger.
3	Scroll/swipe	Use two fingers gestures to control menus in user interface.
3	Opening/closing	Spread the fingers to open the dashboard or close the hand to close the dashboard.

(a) Click   (b) Scroll/swipe

(c-1) Opening   (c-2) Closing
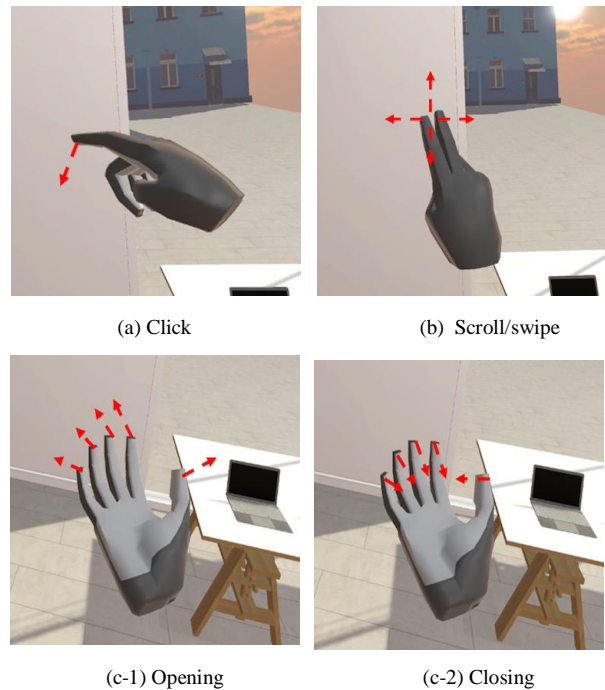
Figure 11. Gestures for interacting with the menu

In a room-scale VR shopping system, the user can walk in the physical walking area in the room. However, owing to the size of the room, the pedestrian zone may sometimes be smaller than the VR shopping store. Thus, we design a gesture for the user to change the area in the room-scale VR shopping store to meet this need. Table VI and Figure 12 show the change in area gesture. To accomplish the area change gesture, one must stretch the index finger and thumb. In the system, when the user intends to change the position of the VR shopping store, one can point the position on the virtual floor with this special hand gesture and use the thumb to click, and the index finger to inform the system where one intends to go.

TABLE VI.   FUNCTIONS OF THE GESTURES FOR CHANGING AREA

Level	Gesture	Function
3	Changing area	Use index finger to point at a new position on the floor and make thumb click index finger, then view in VR will move to the new position.
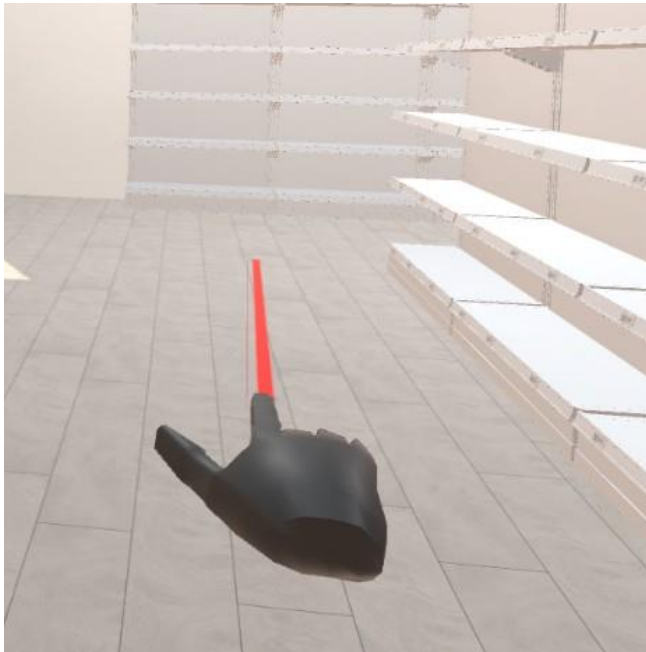
Figure 12. Gesture for changing area in VR environment

VI.   SYSTEM IMPLEMENTATION

In the room-scale VR shopping system, we use HTC Vive as the VR device. The HTC Vive has a head-mounted display (HMD), two controllers, and two base stations. The HMD offers dual 3.6-inch screens with 1080 x 1200 pixels per eye. The refresh rate is 90 Hz, and the field of view is 110 degrees. The controllers have five buttons: multifunction trackpad, grip buttons, dual-stage trigger, system button, and menu button. The two base stations achieve room-scale tracking of HMD and controllers.

We use Leap Motion [17] as the depth sensor to recognize the hand gesture. Leap Motion can track the coordinate and the fingertip and center of palm rotations and transfer the data to the VR system. In the VR system, Leap Motion is mounted on the HMD, as shown in Figure 13.

The VR device and depth sensor require a PC to work accordingly. We use the Unity 3D as the software to build the room-scale VR system. With the Unity 3D, data is processed from Leap Motion and design the virtual shopping environment.

In the room-scale VR shopping system, the user can walk in his own room. Therefore, it is necessary to track the head movement of the user in the 3D space of the room. HTC Vive provides the base stations tracking sensor. We place the two base stations in the two corners of the room.
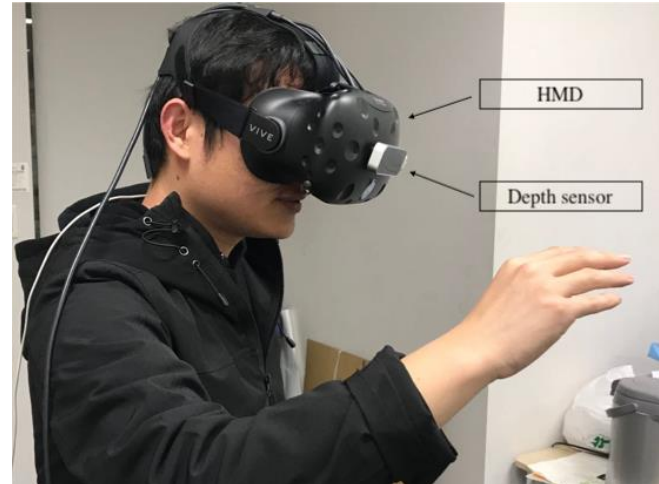
Figure 13. HMD and depth sensor

VII.   GESTURE RECOGNITION

We use Leap Motion as the depth sensor to track the hand. Leap Motion tracks the joints, fingertips, and palm center of the hand of the user. In addition, Leap Motion records the positions of these important points in the hand of the user in every frame.

With the original position data, we use Machine Learning methods to recognize the hand shapes. By combining the hand shapes and motion, we achieve recognition of the gestures that we design for the room-scale VR shopping system.

A.   Hand Shape Recognition

First, we need to confirm how many hand shapes need to be identified because different hand gestures have the same handshape. For example, drag, holding, and rotation gestures share the same handshape. Their differences lie in the palm movement. As we have introduced all 14 gestures, we summarize the following hand shapes that we recognize, also as shown in Figure 14.

The system should be able to distinguish natural hand movements from interaction movements. Therefore, we need to recognize the natural hand shape. There are nine hand shapes that the system needs to realize.

Below is the relationship between the 14 hand gestures and the nine hand shapes:

a. OK handshape (including one gesture): OK gesture.

b. Pointing handshape (including three gestures): pointing gesture, NO gesture, click gesture.

c. Extending the handshape (including five gestures): push/pull, waving, holding, drag, and rotation gestures.

d. Grab handshape (including one gesture): grab gesture.

e. Zoom handshape (including one gesture): zoom in/out gesture.

f. Scroll/swipe handshape (including one gesture): scroll/swipe gesture.

g. Opening/closing handshape (including one gesture): opening/closing gesture.

h. Changing area handshape (including one gesture): changing area gesture.

i. Natural hand shape: the hand shape when the user move the hands without interaction intention.

(g) Opening/closing   (h) Changing area

(i) Natural hand shape

Figure 14. The nine hand shapes

(a) OK hand shape   (b) Pointing hand shape
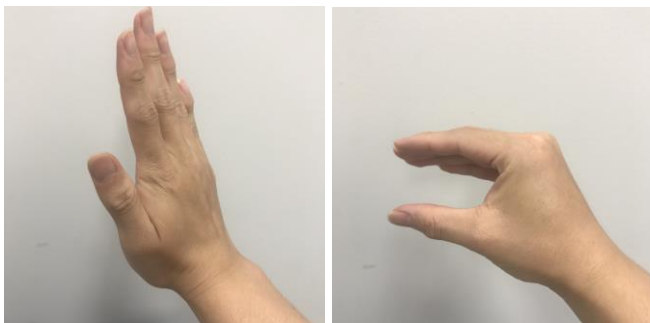
(c) Extending handshape   (d) Grab handshape

(e) Zoom handshape   (f) Scroll/swipe

Figure 15. Key Points: two blue points represent palm center and wrist joint; red points represent the endpoints of bones in the hand

The support vector machine (SVM) method is employed to learn the nine hand shapes and to accomplish the multi-label classification method of the system. We utilize the open-source software, libsvm-3.22, in the VR store [18]. There are four steps for multi-label classification:

1. data collection
2. data normalization and scale
3. model training
4. predicting

(1) Data Collection

From a hand, we capture the endpoints of bones and palm center as "feature points" to describe hand structure [16]. As shown in Figure 15, we track each feature point position data in every frame of the VR environment.

(2) Data Normalization and Scale

We calculate other positions of feature points relative to the palm center. Data normalization follows these steps:

1. Move positions to make palm center on the origin coordinate.

2. Rotate the points to make palm parallel to the x-z axis plane.

3. Rotate the points around the y coordinate axis to make the palm point the z-axis.

Then we scale the data to [-1, 1].

(3) Model Training

The third step of handshape recognition is model training. The objective is from the feature points to recognize the nine hand shapes. Various SVM models have supervised learning strategies with associated learning algorithms that analyze data for classification and regression analysis. We selected the Classification SVM Type 1 (i.e., C-SVM classification) to the system. There are three steps to train with the data and obtain a classifier model:

1. Capture 50 group coordinates of the key points for every hand shape.

2. Normalize and scale the data and obtain nine group training sets.

3. Through the training sets, obtain the multi-label classifier model.

The model training part is preparation work for the system. The multi-label classifier model is employed to realize handshapes in time for the system.

(4) Prediction

After obtaining the multi-label classifier, we utilize it in the VR shopping system to recognize the nine hand shapes. When a user observes the room-scale VR shopping store, the user can freely move the hands. The depth sensor tracks the hands and obtains a group of original data set in every frame. Subsequently, the multi-label classifier predicts the result. The predicted result informs the system which handshape the user is performing. If the hand shape is a natural handshape, the system will disregard it. Conversely, the system acknowledges the command and interacts accordingly.

We put a hand above the Leap Motion and perform the nine handshapes 100 times, respectively. We record the data and test the accuracy of the method, shown in Table VII.

TABLE VII.   THE ACCURACY OF RECOGNITION THE NINE HAND SHAPES

Handshape	OK	Pointing	Extending
Accuracy	92%	94%.	94%
Handshape	**Grab**	**Zoom**	**Scroll/swipe**
Accuracy	92%	93%.	93%
Handshape	**Opening/closing**	**Changing area**	**Natural**
Accuracy	94%	93%.	95%

B. Motion Detection

The system is also required to detect motion because gestures are defined by both hand shape and motion together.

Once getting the hand shape, the system begins the motion detection. The system calculates the hand data in each frame. For different hand shapes prediction, the system detects hand center or different fingertips to realize gestures.

Based on the labeled handshapes from 1 to 9, there are nine situations to recognize the movement.

Situation 1: for OK hand shape, a level 1 category gesture, no need for motion recognition is required.

Situation 2: for the pointing hand shape, if the system detects two hands are in pointing hand shape, the system requires to detect the positions of two index fingertips. If the two index fingertips are close to each other, it indicates the user is performing NO gesture. Conversely, if one hand is in pointing hand shape, the system detects the direction and motion of the index finger to select a target or click a button.

Situation 3: for extending hand shape, the strategy is more complex. (a) if it is detected that the center of the palm is face oriented and gradually approaches the face, it is recognized as a pull gesture; (b) if the palm center is detected as forward-oriented and to move forward, it is recognized as push gesture; (c) if the palm is detected as left-orient and it moves to the left, it is recognized as waving gesture; (d) if the palm is detected as top-oriented, it is recognized as holding gesture; (e) if the palm is detected as forward-oriented and moves on a vertical plane, it is recognized as drag gesture; and (f) if the palm is detected as top-orient and rotating around the palm center, it is recognized as rotation gesture.

Situation 4: for grab hand shape, the system detects the movement of the center of the palm, and the target object follows the movement of the center of the palm.

Situation 5: for zoom hand shape, the system detects the movement of index and thumb fingertips. If the fingertips move away from each other, it is recognized as a zoom-in

gesture; if the fingertips move towards each other, it is recognized as a zoom-out gesture. The movement distance is employed to change the size of the target object.

Situation 6: scroll/swipe handshape; the system detects the movement of the index finger. The movement distance employed to control the menu.

Situation 7: for opening/closing hand shape, the system detects the movement of the index, middle, and thumb fingertips. If they are moving towards each other, it is recognized as the closing gesture; if they are moving away from each other, it is recognized as the opening gesture.

Situation 8: for changing area hand shape, the system detects the direction of the index finger and movement of thumb fingertip. If the index finger points to a position on the floor and thumb fingertip click the index finger, it is recognized as the changing area gesture, and the user moves to the target position.

Situation 9: for natural hand shape, the system does not need to detect any motion because the user moves his hands freely in 3D space and does not wish to interact with the system in this situation.

VIII. Apply Gesture Set In Room-scale VR Shopping Environment

We apply the gesture set in VR environment to build an interactive system. We design a typical shopping activity as an example: viewing and buying a laptop.

Firstly, the user can move to the desk with changing area gesture where the laptops are displayed in the room-scale VR shopping environment, as shown in Figure 16(g). In this situation, the desk is distant from the user.

The user then selects a laptop with the pointing gesture. Once selected, the laptop performs a bounce animation, as shown in Figure 16(a). The user can manipulate the laptop with the hold gesture, as shown in Figure 16(b). After that, he can view the detailed information of the laptop with zoom in/out gesture, open/close gesture, and scroll/swipe gesture, as shown in Figure 16(c), Figure 16(d), and Figure 16(e). Finally, the user can perform an OK gesture to inform the system of the decision to choose it, as shown in Figure 16(f).

By using the gestures we designed, we emulate typical shopping activities in a room-scale VR shopping environment. The system provides the shopping activity process similar to daily shopping activities with physical hand gestures.

(a) Pointing a product (b) Hold the product

(c) Zoom in/out the product (d) Use Open/close gesture

(e) Use scroll/swipe gesture (f) Use OK gesture

(g) Use changing area gesture

Figure 16. The gestures used in shopping activities

IX. Evaluation

Five students were invited to use the room-scale VR shopping system. The ages of the students were 19 - 27. They performed the nine handshapes 50 times respectively. We collected the data and used the SVM method to find the error of the handshape classification.

TABLE VIII.   ERRORS OF CLASSIFICATION OF NINE HAND SHAPES FOR EVERY USER FOR 50 TIMES

Handshape	User1	User2	User3	User4	User5
OK	1	0	0	2	1
Pointing	0	0	1	2	1
Extending	3	2	4	5	4
Grab	1	2	2	3	3
Zoom	0	1	1	3	1
Scroll/swipe	1	1	1	2	1
Opening/closing	0	0	1	1	0
Changing area	1	1	1	4	2
Natural	4	3	4	5	4

TABLE IX.   THE ACCURACY RATE OF HANDSHAPE RECOGNITION OF 5 USER

User	Amount	Accuracy	Accuracy Rate
1	450	439	97.56%
2	450	440	97.78%
3	450	435	95.56%
4	450	423	94.00%
5	450	433	96.22%

Table VIII shows the classification errors of every handshape for every user. From Table VIII, we can see that extending hand shape and natural hand shape have relatively higher error rates because these two hand shapes are relatively similar.

Then we can obtain the accuracy rate when a user performs handshapes in the system, as shown in Table IX.

X.   LIMITATION

Although we did some experiments to examine the accuracy rate of gestures, the method, including selection gestures and SVMs, still requires improvements. The quality of pattern recognition requires to be thoroughly checked by statistical methods. The number of users involved in the experiment is insufficient to obtain robust experimental results.

In addition, although the method is a solution for immersion problem, it requires training to recognize user gestures. Implementing AI systems that recognize standard human gestures may help eliminate training needs. The problem of the lack of haptic feedback in VR gesture interaction remains.

XI.   CONCLUSION AND FUTURE WORK

In this research, we built a room-scale VR shopping system and proposed a new hand gesture set for the room-scale VR shopping system. We employed the gesture set as an alternative to VR device controllers to solve its limitations in VR shopping activities. We introduced a new gesture classification in the gesture set. Three levels for the classification methods are designed. The gestures in level 1 are static hand posture. The gestures in level 2 are dynamic gestures with motion, and the level 3 gestures are the combination of level 1 and level 2 gestures.

For level 3 gestures, we introduced three categories to classify them: core gestures, gestures for interaction with virtual objects, gestures for interaction with menu and interaction with space. The classification helps us understand the gesture set in the room-scale VR shopping system. In addition, the gesture set, and 3-level classification method can be easily transferred to other VR or AR systems.

To achieve complex gestures recognition, we applied the SVM method in the proposed VR shopping system. In the end, the user could walk around in his room to view the VR shopping store and interact with the system with natural hand gestures. We evaluated the accuracy of the gesture set. The results show that gestures have a high recognition accuracy. This suggests that using gestures to replace controllers in a VR environment has a promising prospect.

In the future, we plan to further improve the room-scale VR shopping system such as by implementing an AI system for recognizing gestures. We will improve the accuracy of gesture recognition and the convenience of interaction with system by looking into the recognition failures. The gesture set can be extended to have more specific features in the system and we need to study more complex context before extending the gesture set such as in a more crowded environment with multiple rows of merchandise, where one might partially obstruct the other. We would like to conduct some experiments to evaluate the efficiency of the proposed gesture set by comparing the proposed system with the system using controllers.

REFERENCES

[1] Chunmeng Lu and Jiro Tanaka. "A Virtual Shopping System Based on Room-scale Virtual Reality," The Twelfth International Conference on Advances in Computer-Human Interactions (ACHI 2013) IARIA, Feb. 2019, pp. 191-198, ISSN: 2308-4138, ISBN: 978-1-61208-686-6

[2] Buy+ the first complete VR shopping experience. https://www.alizila.com/video/buy-first-complete-vr-shopping-experience/. Accessed November 29, 2019.

[3] IKEA highlights 2016 virtual reality. https://www.ikea.com/ms/en_US/this-is-ikea/ikea-highlights/Virtual-reality/. Accessed November 22, 2019.

[4] ShelfZone VR shopping experience by inVRsion. http://www.arvrmagazine.com/shelfzone-vr-shopping-experience-by-invrsion/. Accessed November 22, 2019.

[5] Serrano, Berenice, Rosa M. Baños, and Cristina Botella. "Virtual reality and stimulation of touch and smell for inducing relaxation: A randomized controlled trial." Computers in Human Behavior, vol. 55, pp. 1-8, 2016, doi: https://doi.org/10.1016/j.chb.2015.08.007. Accessed November 22, 2019.

[6] Bhatt, Ganesh. "Bringing virtual reality for commercial Web sites." International Journal of Human-Computer Studies, vol. 60, pp. 1-15, 2004, doi: http://dx.doi.org/10.1016/j.ijhcs.2003.07.002. Accessed November 22, 2019.

[7] Chen, Tian, Zhi-geng Pan, and Jian-ming Zheng. "EasyMall-An Interactive Virtual Shopping System." 2008 Fifth International Conference on Fuzzy Systems and Knowledge Discovery, vol. 4. IEEE, 2008, doi: http://dx.doi.org/10.1109/FSKD.2008.124.

[8] Lee, Kun Chang, and Namho Chung. "Empirical analysis of consumer reaction to the virtual reality shopping mall." Computers in Human Behavior, vol.24, pp.88-104, 2008, doi: http://dx.doi.org/10.1016/j.chb.2007.01.018.

[9] Verhulst A, Normand J M, Lombart C, et al. "A study on the use of an immersive Virtual Reality store to investigate consumer perceptions and purchase behavior toward non-standard fruits and vegetables." 2017 IEEE Virtual Reality (VR). IEEE, pp.55-63, 2017, doi: 10.1109/VR.2017.7892231. Accessed December 7, 2019.

[10] Speicher, Marco, Sebastian Cucerca, and Antonio Krüger. "Vrshop: A mobile interactive virtual reality shopping environment combining the benefits of on-and offline shopping." Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies, vol 1, No. 3, Article 102, doi: http://dx.doi.org/10.1145/3130967. Accessed November 22, 2019.

[11] Lin, JJ-W., et al. "Effects of field of view on presence, enjoyment, memory, and simulator sickness in a virtual environment." Proceedings ieee virtual reality 2002, pp. 164–171, doi: http://dx.doi.org/10.1109/VR.2002.996519.

[12] Garg, Pragati, Naveen Aggarwal, and Sanjeev Sofat. "Vision based hand gesture recognition." World Academy of Science, Engineering and Technology, vol 49, pp. 972-977, 2009. http://waset.org/Publications?p=25. Accessed November 22, 2019.

[13] Wachs, Juan Pablo, et al. "Vision-based hand-gesture applications." Communications of the ACM, vol 54, pp. 60-71, 2011, doi: http://dx.doi.org/10.1145/1897816.1897838. Accessed November 22, 2019.

[14] Boussemart, Yves, et al. "A framework for 3D visualisation and manipulation in an immersive space using an untethered bimanual gestural interface." Proceedings of the ACM symposium on Virtual reality software and technology. ACM, pp. 162-165, 2004, doi: http://dx.doi.org/10.1145/1077534.1077566. Accessed November 22, 2019.

[15] Karam, Hani, and Jiro Tanaka. "Two-handed interactive menu: An application of asymmetric bimanual gestures and depth based selection techniques." International Conference on Human Interface and the Management of Information. Springer, Cham, pp. 187-198, 2014, doi:http://dx.doi.org/10.1007/978-3-319-07731-4_19. Accessed November 22, 2019.

[16] Lu, Chunmeng, Li Zhou, and Jiro Tanaka. "Realizing Multi-Touch-Like Gestures in 3D Space." International Conference on Human Interface and the Management of Information. Springer, Cham, pp.227-239, 2018, doi: http://dx.doi.org/10.1007/978-3-319-92043-6_20. Accessed November 22, 2019.

[17] Leap Motion, 2018. https://www.leapmotion.com/. Accessed November 22, 2019.

[18] LIBSVM -- A Library for Support Vector Machines, 2016. https://www.csie.ntu.edu.tw/~cjlin/libsvm/. Accessed November 22, 2019.

Approaches to Develop and Implement ISO/IEC 27001 Standard - Information Security Management Systems: A Systematic Literature Review

Daniel Ganji

Centre for Secure, Intelligent and Usable Systems (CSIUS)
University of Brighton
Brighton, UK
d.ganji2@brighton.ac.uk

Christos Kalloniatis

Privacy Engineering and Social Informatics Laboratory
Department of Cultural Technology and Communication
University of the Aegean
Greece
chkallon@aegean.gr

Haralambos Mouratidis,
Saeed Malekshahi Gheytassi

Centre for Secure, Intelligent and Usable Systems (CSIUS)
University of Brighton
Brighton, UK
h.mouratidis@brighton.ac.uk
m.s.malekshahi@brighton.ac.uk

Abstract—**This systematic literature review intends to determine the extent to which contribution is available to assist organisations and interested parties to understand better or comply with the requirements of the ISO/IEC 27001 international standard, known as Information Security Management Systems (ISMS). The primary aim of this paper is to explore the current literature in the ISMS as specified by the ISO/IEC 27001 standard, aiming to provide a mapping of their contributions with the requirements of the standard. An objective of this study is to explore the ways in which the literature addresses the requirements of the ISMS. This study uses semi-quantitative analysis in order to gain insights into the concepts and techniques around the ISMS and to systematically obtain data to help with identifying the research gaps. One of the findings of this review is to encourage to benefit from available literature and to develop an ISMS to promote their corporate compliance with a well-established standard. The most striking result from the review is that the majority of approaches proposed by scholars between 2005 to 2018 are with limited support in adopting the information security management system. Another important finding is that almost all available approaches fundamentally lack the motivation to focus on the analysis and application of the ISMS with no single study enable organisations to adopt the ISO/IEC 27001 standard.**

Keywords–ISO/IEC 27001; information security management systems; PDCA; requirements engineering; information security risk management.

I. INTRODUCTION

In the new global economy, organisations face tougher pressure in securing their internal and external information. Some of these pressures are through national and international laws and regulations, interested parties expectations, and organisational requirements to safeguard their business secrets from their competitors. It is a compelling task for organisations to meet the security requirements and take the necessary actions to implement and satisfy their security objectives [1]. In contrast, the continual change in technology, management use of technology and the impact on business success makes the management information systems an exciting topic in organisations [2].

To date, there has been no solid evidence to absolute security and protection, however, there are available security frameworks and techniques to promote the best practices in managing information security. Organisations need to prepare towards sophisticated approaches considering security and its associates under one interconnected application to successfully manage confidentiality, integrity, and availability of information assets. The numbers of security breaches are getting bigger and invaders are getting smarter in ways to exploit security vulnerabilities. Conventional and outdated managing of information security does not answer the needs of the current structure [3] [4]. Experts believe that more than 90% of successful cyber attacks could have been prevented by the technology available at the time [5].

Information Security Management System (ISMS) as defined by the International Organization for Standardization (ISO) and the International Electrotechnical Commission (IEC) 27001 is an international standard to provide requirements for establishing, implementing, maintaining and continually improving an information security practices in organisations. ISO/IEC 27001 is integrated part of the organizations processes and overall management structure and that information security is considered in the design of processes, information systems, and controls. The standard is applicable to all organisations, regardless of their type, size, or nature [6] and it constitutes a certifiable standard and is widely used with steady growth in a number of adoptions [7]. The standard provides mapping for establishing, implementing, maintaining and continually improving an information security management system or alternatively known as Plan-Do-Check-Act (PDCA) model.

It is a strategic decision for an organisation to adopt ISMS and to preserve the confidentiality, integrity, and availability of information by applying risk management process and giving confidence to interested parties that risks are adequately managed. ISMS is composed of processes, policies, and resources that can be used to systematise the security demands of an organisation. Organisations understand that it is in their interest to follow some type of internationally recognised reference framework to create environments for information security management systems rather than doing it ad hoc [8].

The primary aim of this systematic review is to investigate in detail the available software engineering techniques on the ISMS that enable organisations to comply with the require-
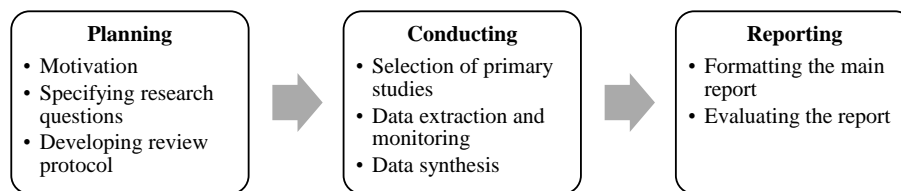
Figure 1. Summary of the phases in the systematic literature review

ments of the ISO/IEC 27001 standard. Additionally, the review explores the strengths and limitations of the current literature to establish discussion for future work.

The remaining part of the paper proceeds as follows: Section II describes the methodology used as part of this review including the phases of the review and data analysis. The results of the review are reported in Section III. Section IV discusses the findings of the review, focusing on the three research questions set out in the next section. Lastly, Section V provides a brief overview of the key findings, making suggestions for future work.

II. METHOD

This study conducted in the form of a systematic literature review by employing Guidelines for Performing Systematic Literature Reviews in Software Engineering introduced by Kitchenham et al. [9] [10] and Webster et al. [11].

The review involved a series of activities divided into three phases are shown in Figure 1. The steps in the review method documented below.

Planning phase: The initial step sketched the need for undertaking this review by considering all existing information about the ISO/IEC 27001 standard and software engineering in a thorough and impartial manner. The second step of the planning was to specify the research questions by considering the types and structure of the questions as discussed in Section II-A. The last part of the planning phase was to develop a review protocol to specifies the methods that will be used to undertake the review and reducing the possibility of a bias. The components of our review protocol include study selection procedures explained in Section II-B, study selection criteria defined in Section II-C, data extraction strategy set out in Section II-D, and synthesis of the extracted data explained in Section II-E.

Conducting: This phase implemented the steps identified in the research protocol from the former phase. The initial step identified the primary studies to provide direct evidence about the research questions. Next, to accurately recorded the information obtain from the primary studies. Finally, a descriptive synthesis of the primary studies developed to provide a summary of the results in Section III.

Reporting: The last phase involved the writing of the review findings obtained from the results section are summarised in Section IV.

A. Motivation and Research questions

The research to date from the industry and academia tend to focus on the overall description of the standard and such expositions are unsatisfactory because little is being contributed to the practicality of the ISMS structure. The generalisability of much-published research on the standard is insufficient for organisations aiming to implement the standard.

IT Governance, a provider of IT compliance solutions to organisations released an annual survey [12] centred around the experience and implementation challenges of the ISO/IEC 27001 for organisations in 2016. The investigation of 250 information security professionals from 53 countries who participated in the survey were mostly certified or working towards certification (80%). 71% of respondents received either regular or occasional requests to provide the ISO/IEC 27001 certification from clients or when proposing for new business. By providing compliance to a globally known standard, certification significantly reduces the need for repeated client audits. The survey also found that a third of all respondents were concerned about understanding the requirements of the standard and 28% considered the creation and managing the standard documentation a challenging task. Other substantial challenging tasks were conducting the information security risk assessment and identifying the required controls for 22% and 14% of the respondents respectively.

From the commercial aspect, it is a rather difficult and costly task to identify the resources required to implement, measure, and manage information security. From an academic perspective, ISMS have mostly drawn from the views of practitioners [13] and our literature review indicates that ISMS has not been particularly attractive in academia with a lack of research and approaches are egregious. Management systems on information security received very limited observation and research from the academic community despite the high interest from organisations in particular for IT, operational and compliance audits [14].

The purpose of this review is to systematically evaluate and measure the current literature in compare with the requirements of the standard and gain further understanding of the gap in the literature. The review sought to answer the following research questions:

RQ1. What are the software engineering approaches that organisations could use to apply or implement the requirements of the ISMS as defined by the ISO/IEC 27001 standard? We aim to identify the software engineering techniques and tools to assist interested parties, such as information security officers, compliance managers, and top management in organisations to adopt and comply with the requirements of the ISO/IEC 27001 standard.

RQ2. What are the scholarly contributions to the literature since the introduction of the ISO/IEC 27001 standard in 2005? Research into a good security practice has a long history back to 1989 when a set of internationally recognised security evaluation criteria was developed by the Department of Trade and Industry (DTI) Commercial Computer Security Centre

(CCSC). This was further developed by British Standard Institute to British Standard BS7799-1 in 1995 and later adopted by ISO/IEC 17799 as Code of practice for Information Security Management in 2000; this document was reproduced to ISO/IEC 27001 ISMS requirements in 2005. The 2005 version of the standard was extensively revised in 2013, it became generic with more flexibility and some controls were added or changed in the new and current version of the standard. Part of the aim of this review is to trace the development of the literature within the life of the standard from its introduction in 2005.

RQ3. What are the limitations of the current research?
Another aim of this paper is to critically analyse the effects of the current literature in comparison with the requirements of the standard and whether the literature provides sufficient contribution to facilitate the use of the standard for organisations.

B. Search process

Each journal and conference proceedings were reviewed and assessed by the first author, however, the papers that addressed literature of any type identified as included or excluded were discussed with the other researchers. The researcher responsible for searching the journal or conference applied the detailed inclusion and exclusion criteria to the relevant papers. The automated search strategy was followed in our research to identify the primary studies. The electronic libraries used were:

- Google Scholar
- IEEE Xplore
- Springer
- Science Direct
- Research Gate
- British Library EThOS
- ACM Digital Library
- Abstracts in New Technologies and Engineering
- Web of Science

As part of the literature studies, certain keywords and synonyms were established and included in the research. We worked on keywords and terms that these studies use to specify essential concepts of relevance to ISMS. For the retrieval in the digital libraries, a sophisticated search string was constructed using Boolean ANDs and ORs. The string given below was derived and taken as a basis to apply to the title, keywords, and abstracts of publications:

(('iso/iec 27001 standard' OR 'information security management systems' OR 'isms' OR 'information security standard' OR 'security standard') AND ('requirements engineering' OR 'compliance engineering' OR 'security requirements engineering' OR 'software engineering'))

The above search strings were assessed using the applicable elements from Peer Review of Electronic Search Strategies (PRESS) checklist by McGown et al. [15]. The validation results obtained from the PRESS assessment are set out in Table I. Some electronic libraries did not provide advanced search options that allow for the use of the search string as is. For these sites, we either extended the context of the search or separated the search into several sub-searches preserving the

initial search context. The selection of primary studies was governed by the inclusion and exclusion criteria.

TABLE I. Elements from PRESS checklist

No	PRESS Element	Result
1	Are the search concepts clear?	Yes
2	Does the search string match the research question?	Yes
3	Are there any mistakes in the use of Boolean or nesting?	No
4	Are the subject headings relevant?	Yes
5	Are the subject headings missing?	No
6	Are any subject headings too broad or too narrow?	No
7	Does the search miss any synonyms?	No
8	Does the full term included for the abbreviation used?	Yes
9	Are there any spelling errors?	No
10	Are any filters used appropriate for the topic?	Yes
11	Are any potentially helpful limits or filters missing?	No

C. Inclusion and exclusion criteria

Peer-reviewed articles on the the following topics were included:

- An article published between 01 Jan 2005 and 30 June 2018: we wanted to cover the years that both versions of the standard published in 2005 and 2013, hence, it is fair to cover from the start of 2005 until the current date.

- An article should discuss the search string described in Section II-B.

- An article should propose a software engineering technique in addressing the standard: the aim of this paper is to capture the contributions from the field of software engineering.

Articles on the following topics were excluded:

- An article that is not written in English.

- White papers or informal articles: not peer-reviewed papers or articles, which provide a plain description of the standard rather than purposing a technicality were excluded.

- Duplicate reports of the same study: when several reports of a study exist in different journals the most complete version of the study was included in the review.

D. Data collection

This review does not claim to have captured every approach within the ISMS, however, the aim of this review is to have a holistic comprehension of the current state of the art in the ISMS. We recognise there could be a number of other related approaches that consider other ISMS methodologies such as ISACA COBIT or NIST Cybersecurity Framework, however, the intention of this paper is ISO/IEC 27001 standard and to achieve a fairly detailed conclusion within this topic.

The information extracted from the selected studies must reflect our research questions and indicate a desirable contribution towards the ISO/IEC 27001 standard. The initial studies of 285 papers were converged by learning their meta-data including title, abstract, keywords, and conclusion. A total of 95 papers met our objectives and aims of this review, which led us to further investigate the full text of a study. Finally, 21

papers were selected as primary studies for in-depth evaluation and participation in our review paper.

The order of reporting the primary studies in the next section is in chronological order and for the purpose of fairness and accuracy, the same amount of information about each selected study was extracted. The data extracted from each study were:

Approach title: This is a proposed title by the authors of a primary study for his/her approach or contribution. If a title was not available then we referred to the first author's full name.

Year of publication: The year when the paper was published. If a paper was published in several different sources both dates were recorded and the first date was used in any analysis.

Type: Each primary study was categorised into two terminologies including Framework or Method, the definition used for each is as follow:

- Framework: This is a process or conceptual layered structure intended to serve as a support or guideline for the building of something useful [16].

- Method: It refers to the methods the researchers use in performing an operation [17].

Scope: The scope equally measures the contribution of a study towards the PDCA model. The four stages include:

- Plan: Establish the ISMS policy, objectives, processes and procedures relevant to managing risk and improving information security.

- Do: Implement and operate the ISMS policy, controls, process and procedures.

- Check: Assess and measure process performance against ISMS policy.

- Act: Maintain and improve the ISMS by taking corrective actions where nonconformity occurs.

Findings and practical implications: This term refers to analysis, discussion, results, and identification of outcomes and implications for practice in the primary studies. In case of duplicate publications, the most completed paper among those was used by referring to the versions of the report to obtain all the necessary data.

E. Data analysis

A set of 22 criteria as described in Table II were excerpted from the clauses and sub-clauses of the ISO/IEC 27001:2013 standard to compare and evaluate the identified studies.

The standard specifies the requirements for establishing, implementing, maintaining and continually improving an ISMS within the context of the organisation. Excluding any of the requirements is not acceptable when an organisation claims conformity to this standard, hence, a similar approach is used to measure the level of fulfilment to all requirements of the standard by each identified study.

The same definition for each criterion as specified in the standard [18] is followed, to allow the established uniform description used in our review and to avoid misinterpretation or misjudgement. These criteria were selected from the current version of the standard published in 2013, however,

it is recognised that majority of the literature was published prior to 2013, therefore, a formal mapping [19] of ISO/IEC 27001:2013 clauses to ISO/IEC 27001:2005 version were used to ensure that papers published prior to 2013 are not disadvantaged in comparison with papers published post 2013.

The order in presenting the criteria do not reflect their importance or imply their implementing order; the list items are enumerated for reference purpose only.

III. RESULTS

The following summarises the result of our review from the selected studies under the keywords that this research interested to investigate.

Chang and Ho proposed a model [20] [21] to explore the influence of organisational factors on the effectiveness of implementing the BS7799 (replaced by ISO/IEC 27001) standard. The findings defined four factors that could cause a serious impact on the success of the implementation of the information security management, they included IT competence of business managers, environmental uncertainty, industry type, and organisational size. The impact of these factors could be varied between any types of organisations. The findings indicate large organisations may benefit more in implementing information system security standards since they are more depended on formalisation and standardisation than small companies and have a greater amount of assets. Their studies were limited as only targeted 59 organisations in Taiwan but it was expected to have a similar result for another region too.

Mellado et al. proposed Security Requirements Engineering Process (SREP) [22] [23] to incorporate security requirements such as Common Criteria (ISO/IEC 15480) into the software life cycle model in a structured process. SREP used a collection of standards, processes and activities for the development of secure information systems under a systematic approach. The framework was made up of nine activities known as micro-process to form the security requirements engineering, as well as the external and visible artefacts that involve the activities. The activities included the determination of the security vision, understanding of the stakeholders, the identification of the vulnerabilities and assets, identification of security objectives and threats, risk assessment, the elicitation-prioritisation- inspection of security requirements and the repository improvement.

Anwar et al. proposed Preventive Information Security Management (PrISM) [24] system, a model to advance the security assurance and risk handling process in an ISMS with intrusion prevention capabilities. PrISM developed a network security solution including a number of services and functionalities, such as intrusion, detection and prevention capability, integrity checks, incident management and managerial reporting. The above could be incorporated in a single control panel to enable the integration, summarising and linking all the tools and functionalities together. This could assist with automating incident handling and other tasks, which could minimise the operational risks within organisations using comprehensive security monitoring.

Fenz et al. proposed OntoWorks [25] [26], which is an ontological mapping of the ISO/IEC 27001 standard supporting the certification process. Authors proposed a framework to use ontological data and enable users to access, visualise,

TABLE II. Criteria from the ISO/IEC 27001 Standard

No	Criterion	Description
1	Organisational context	Define the external and internal parameters and issues affecting the outcome of ISMS.
2	Interested parties	Identify the interested parties and their information security requirements relevant to the ISMS.
3	Determining the scope	Identify the logical or physical boundaries and applicability of the ISMS
4	ISMS	Establish, implement, and continually improve an ISMS under the requirements of the standard.
5	Leadership	Top management to demonstrate leadership and commitment with respect to the ISMS that are compatible with the strategic direction of the organisation.
6	Policy	Establish directions and making references to IS objectives and appropriate to the purpose and context of the organisation.
7	Roles	Top management to assign and communicate the responsibilities and authorities relevant to information security for reporting performance of the ISMS within the organisation.
8	Risk & opportunities	Systematically determine the potential risks and opportunities that may be involved in a projected activity or undertaking.
9	Information security objectives	Define measurable information security objectives.
10	Resources	Identify the resources needs to manage the ISMS.
11	Competence	Identify the necessary ability of a persons knowledge and skills doing work under its control that affects information security performance.
12	Awareness	Persons working under the organisation's control to be aware of the information security policy and their contribution to the effectiveness of the ISMS.
13	Communication	Apply internal and external communication process relevant to the ISMS.
14	Documented information	Create, update, and control documented information required by the standard and necessary for the effectiveness of the ISMS.
15	Operational planning	Plan, implement and control the process needed to meet information security requirements including risk and opportunities, and information security objectives.
16	IS risk assessment	Perform security risk assessment.
17	IS risk treatment	Implement information security risk treatment.
18	Monitoring & measurement	Evaluate the information security performance and its effectiveness.
19	Internal audit	Conduct regular internal audits and systematically evaluate the effectiveness of the implemented and maintained ISMS.
20	Management review	Top management to review the organisation ISMS at planned intervals to ensure its continuing suitability, adequacy and effectiveness.
21	Nonconformity & corrective action	React and evaluate nonconformity occurrences, review and deal with appropriate corrective actions.
22	Continual improvement	Recurring activity to continually improve the suitability, adequacy and effectiveness of the ISMS.

and reason on ontological data. Their contribution helped for audit preparation and rule-based compliance checks regarding ISO/IEC 27001 controls. As some of the operations delivered as partial automation, this will increase the automation process within the certification process, resulting in saving costs and resources. Fenz et al. [27] later proposed security ontology to be used to increase the efficiency of the compliance checking process by introducing a formal representation of the ISO/IEC 27002 standard.

Mellado et al. proposed Security Requirements Engineering Process for Software Product Lines (SREPPLine) [28], [29], this was a solution for managing security requirements at an early stage of the product line development driven by security standards. This framework was structured management of the security requirements to facilitate the conformance of the software product line products to relevant security standards such as ISO/IEC 27001 and ISO/IEC 15408. The proposal consisted of two sub-process including the product line security domain engineering and the product line security application requirements engineering. These sub-processes responsible for four phases of requirements engineering, such as requirements elicitation, requirements analysis and negotiation, requirements documentation, and requirements validation and verification. Mellado et al. [30] later used Secure Tropos framework for Software Product Lines requirements engineering for elicitation of security requirements and their analysis on both a social and technical dimensions.

Boehmer proposed a methodology [31] [32] to measure the effectiveness of the implementation and operation of an ISMS in organisations. The methodology delivered a solution to form an assessment through audits checking of the internal controls. Internal controls included administrative controls, physical controls, and technical controls.

Mayer proposed Information System Security Risk Management (ISSRM) [33] [34] [35], providing a reference conceptual model for security risk management. The author proposed a model-based approach for ISSRM, applicable since the early phases of IS development. The work focused on the modelling support to such an approach, by proposing a domain model for ISSRM. The work defined a reference conceptual model for security risk management and enhancement of the domain model with the different metrics used in a risk management method. Further, the authors developed a proposal of the Secure Tropos language and a process to use the extension in the frame of risk management.

Ekelhart et al. proposed AUtomated Risk and Utility Management (AURUM) [36] [37], a risk management methodology to support the NIST 800-30 risk management standard. The methodology focused on the risk management approach by conducting various techniques such as questionnaires, on-site interviews, document reviews, and automated scanning tools to gather the required information under an ontological framework. AURUM provides risks assessment management by understanding the organisation characterisation followed by vulnerability identification, threat identification, risk likelihood determination, control analysis, control recommendations with appropriate controls, cost/benefit evaluation, impact analysis, modelled and taken from best practise standards such as the IT Grundschutz. This is a methodology for supporting information security risk management through modelling the organisation's assets within an ontological framework.

Valdevit et al. proposed an approach [38] [39] on how to adopt ISO 27001 on SMEs and their specific needs in implementing the ISMS. They developed their approach to knowledge gained in SMEs for several years in several disciplines and sectors. This was an approach where researchers

and practitioners work together, towards a number of activities including problem diagnosis, active intervention, and reflective learning. Authors described their approach as a "blend of theoretical reviews and experiments".

Hensel and Lemke-Rust proposed an approach [40] of Braun [41] to business engineering was chosen for the integration of ISO/IEC 27001 into an enterprise architecture. Authors integrated an ISMS into a systematic business engineering. The approach consisted of four layers such as strategic layer considered the internal and external requirements of an organisation and its strategic alignment; organisation layer considered the overall organisation process vision and defines the roles and responsibilities of the ISMS; the information system layer considered the information assets and information architecture of the organisation including software component and platform view; infrastructure and technology layer considered the infrastructure used for conducting a risk analysis of an ISMS.

Schneider et al. proposed Heuristic Requirements Assistant (HeRA) [42], an assistant tool to enable the identification and analysis of security requirements by applying experience-based tool rather than dependency on experts. An approach to provide knowledge about security best practises to developers and designers with limited experience. This approach is based on modelling the flow and enabling the stakeholders to exchange, learning and reusing relevant experiences about security requirements at the project requirements level.

Muller et al. introduced a tool [43] [44] to supports cloud service providers and consumers under a security management platform. A Security Management Platform (SMP) to specify the security requirements and measure the effectiveness of implemented controls for cloud service providers and consumers to conjointly manage information security. The system management platform consisted of three steps: service provider and consumer identify the security requirements for a cloud service in order to prepare a specific service level agreement based on agreed requirements, service provider manage and maintain the implementation and operation of security controls in a traceable and transparent manner, service provider is responsible to measure the specified requirements, identified in the first step and periodically to generate reports and incident reports about implemented controls to stakeholders.

Gillies proposed 5S2IS [45], an approach to facilitate SMEs to implement and comply with ISO/IEC 27000 standard. The proposed approach developed a two-dimensional matrix with the use of ISO/IEC standard and the Capability Maturity Model (CMM). It included draw up a plan to understand the organisation expectation and achieve the ISMS, define policies and processes to reach the organisation goals, identify the non-compliances with the goals through measurement, analyse and identify the growth and improvement of performance through monitoring, embed the ISMS in the organisation and plan to attain for certification if applicable.

Susanto et al. proposed Integrated Solution Framework (I-SolFramework) [46] [47] [48] to assesses the readiness level of an organisation towards the implementation of ISO 27001. The framework offered e-assessment and e-monitoring to analyse and performed an assessment of the readiness level of ISO 27001 implementation. E-assessment measure ISO 27001 parameters based on the framework; it consisted of six layers component including organisation, stakeholder,

tools and technology, policy, culture and knowledge with 21 controls. It helped to validate the ISO/IEC 27001 parameters through an analytical interface such as histogram, charts and graphs, provided by a framework.

Montesino et al. proposed Security Information and Event Management (SIEM) [49]. A framework to enable the organisation to evaluate their compliance with IS standards and their implementation effectiveness by automatically generating ISO 27001 based on IT security metrics [50]. Authors findings indicated about 30% of the security controls of ISO/IEC 27001 standard can be automated. SIEM technology consisted of two main functions of security information management system, which handles the collection, reporting and analysis of log data; and security event management, which monitors real-time data and manages incident of security-related events. SIEM based solution is proposed to centralise and incorporate a list of ten automated controls including asset inventory, account management, log management, system monitoring, malware protection, vulnerability scanning and patch management, security configuration assessment and compliance checking, information backup, physical security, incident management.

Azuwa et al. proposed Supervisory Control and Data Acquisition (SCADA) [51] [52], An approach to measure the effectiveness of network security management in SCADA. This method specifically assisted to enable a measurement approach to the effectiveness of ISO/IEC 27004 measurement standard. It initially identifies security controls followed by a risk management approach to develop risk-based requirements and prioritisation of security control implementation. This step included the identification of threats and vulnerabilities and their impacts. The third stage was to develop an effective measurement and metric through questionnaires and interviews, perception and experts knowledge, certified organisations and SCADA owners.

Beckers et al. proposed a methodology [53] [54] to analyse security requirements engineering methods to support the development and documentation of an ISMS according to ISO/IEC 27001. Authors described the aims to improve the result of ISO 27001 implementation through proper establishment and documentation of an ISMS.

Chatzipoulidis et al. proposed a risk management approach [55] called "to be" environment by focusing on analysing threats, evaluating and treating vulnerabilities in the information society. The author described information society as a dynamic information security management system and proposed a concept to enhance the role of e-government to support public administration and cognitive resource for policymakers. The "to be" environment methodology identified risks by characterising the elements of risks and summarising critical threats of cyberbullying and cyberstalking attack patterns; identification of risk by analysing cultural dynamics and assessment of the current and planned controls of the system in place; evaluation of risk by producing a list of critical risks, prioritised based on set criteria; and risk treatment to lessen risks to meet the risk appetite level.

Asosheh et al. proposed a framework [56] for implementing an ISMS within a large-scale enterprise to assist them in identifying related activities in establishing and implementing an ISMS including the risk assessment and treatment procedures. The process consisted of five steps according to

ISO/IEC 27003 implementation guidance such as obtaining management approval for initiating the ISMS project. The steps included a preliminary scope identification and preparing definitions for ISMS and a business plan to have the management approval. Defining ISMS scope, boundaries and ISMS policy, which helps to produce a final document to set the boundaries for the ISMS policy and scopes, conducting information security requirements analysis that aims to identify assets and needs of asset owners, and risk management.

Beckers et al. proposed PAttern-based method for establishing a Cloud specific informaTion Security management system (PACTS) [57] [58] [59] [60]. An approach for creating an ISMS methodology compliance to the ISO 27001 standard cloud environment with a specific interest in legal compliance and privacy. The overview of the methodology was leadership commitment, asset identification, threats analysis, risk assessment, security policies and reasoning, ISMS specification, identify relevant laws and regulations, the definition of compliance controls, instantiating privacy patterns, privacy threats analysis.

Beckers et al. proposed ISMS-CORAS [61] [62], an extension of the COROS method to support the establishment of an ISO/IEC 27001 compliant ISMS. Authors proposed a methodology following CORAS method. CORAS is a risk management methodology based on the ISO 31000 standard, therefore, providing compliance to ISO 31000 standard, consideration of legal concerns tool support for document generation. CORAS-ISMS support security management compliant with ISO/IEC 27001 standard.

IV. DISCUSSION

In this section, we discuss the answers to our research questions described in Section II-A.

RQ1. What are the software engineering approaches that organisations could use to apply or implement the requirements of the ISMS as defined by the ISO/IEC 27001 standard?

An overall description of the primary studies are shown in Table III where appending each study Title, Year of publication, Type of contribution indicating Framework (F) or Method (M), Scope(s) of the PDCA model covered by each study, and depth of fulfilment at each Stage of the Plan, Do, Check, and Act. A summary of each study was described in the previous section.

Our speculation with respect to the PDCA model is that very little attention is given at the Check stage where only five studies out of 21 provided contribution to the relevant part of the standard. Check specifically deals with assessment and measurement process performance against the ISMS.

Act stage tends to have less to almost no contribution where only one study out of 21 identified to address the relevant part of the standard. Act maintains and improves an ISMS by taking corrective actions where nonconformities occur. Interestingly, even some of the proficient concepts like ISMS-CORAS or ISSRM did not target any of the named stages of the standard in their studies.

RQ2. What are the scholarly contributions to the literature since the introduction of the ISO/IEC 27001 standard in 2005?

The chart in Figure 2 depicts the overall fulfilment percentage of each study towards the requirements of the standard in chronological order from 2005 to 2018.

TABLE III. Overall description of primary studies

Title	Year	Type	Plan	Do	Check	Act
Chang, Shuchih Ernest	2006	M	+	-	-	-
SREP	2007	F	+	+	-	-
PrISM	2007	M	-	-	+++	++
OntoWorks	2007	F	-	+	++	-
SREPPLine	2008	F	++	+	-	-
Boehmer, Wolfgang	2008	M	+	+	-	-
ISSRM	2008	M	++	+++	+	-
AURUM	2009	M	+	++	-	-
Valdevit, Thierry	2009	M	+	-	-	-
Hensel, Veselina	2010	M	+	++	-	-
HeRA	2011	M	+	-	-	-
SMP	2011	F	+	-	++	-
5S2IS	2011	F	+	+	++	-
I-SolFramework	2012	F	++	-	-	-
SIEM	2012	F	-	+++	-	-
SCADA	2012	M	-	++	-	-
Beckers, Kristian	2012	M	+	+	-	-
"to be" environment	2013	M	+	++	-	-
Asosheh, Abbass	2013	M	+	+++	-	-
PACTS	2013	M	++	++	-	-
ISMS-CORAS	2013	F	++	+++	-	-

Note:
F = Framework
M = Method
- = Not fulfilled
+ = Partially fulfilled = Number of criteria per scope:
Plan [1-5], Do [1], Check [1], Act [1]
++ = Mostly fulfilled = Number of criteria per scope:
Plan [6-10], Do [2], Check [2], Act [1]
+++ = Fulfilled = Number of criteria per scope:
Plan [11-14], Do [3], Check [3], Act [2]

The trend indicates the current studies are fragmented and it is a challenging task for organisations to benefit from the current literature. Alternatively, they require to apply a number of studies in conjunction with each other that may result to an inconsistent, unmanageable, and intractable output. Whilst the existing work could help with rather smaller sections of the standard and used as a point of reference but they are inadequate to realise the full requirements of the standard.

Our findings suggest that the majority of studies proposed between 2005 to 2018 are incomplete and they mostly provide a partial fulfilment to the requirements of the ISO/IEC 27001 standard. This review provides evidence with respect to a gap in the field of the ISMS.

The graph in Figure 3 reveals that a reasonable quantity of the studies were produced between 2006-2008, after the publication of the first version of the standard in 2005; the attention dropped until around 2010.

Half of the studies carried out between 2011 to 2013 and it appears the consideration to the ISMS was higher prior to the publication of the second version of the standard in 2013 than after. This shows an inconsistent and contradicts association between the first version and the second version of the standard. A possible explanation could be the fact that other standard documents in the family of ISO/IEC 27000 were revised and published between 2011-2013, such as a revised publication of ISO/IEC 27003 in 2010, ISO/IEC 2005 in 2011, ISO/IEC 27006 in 2011, ISO/IEC 27007 in 2011, ISO/IEC 27008 in 2011.

The most striking result to emerge from the data is that the expansion of further research dropped sharply after 2013 and no study was detected after the revised publication of the standard in 2013, which should have caused some spark in
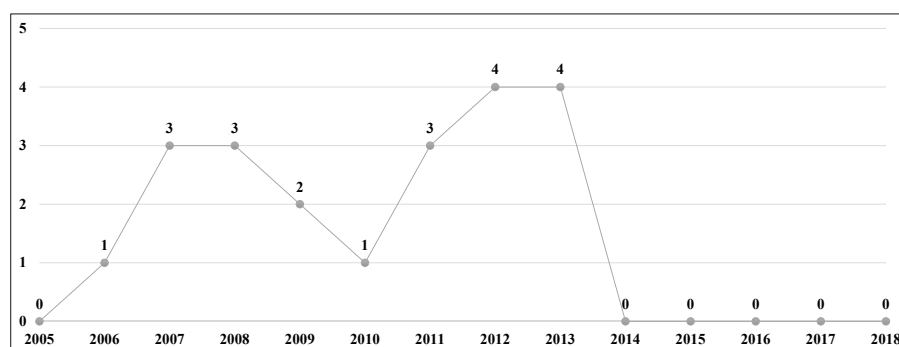
Figure 2. Overall fulfilment of each study towards the requirements of the ISO/IEC 27001 standard in (%)

academia. It is interesting to note that almost all papers (20) were published between 2007 to 2013.

RQ3. What are the limitations of the current research?

Table IV demonstrates a detailed review of all 21 studies and their contribution at each criterion identified in the previous section; It provides the overall strength and limitation of the study. The indicative asterisk (*) in the table denotes the fulfilment of a criterion.

On average, some criteria are shown to have attracted the majority of the literature than others. The areas where significant differences have been found include Organisational context (1), Interested parties (2), Determining the scope (3), Documented information (14), Operational planning (15), and Information security risk assessment (16).
On the other hand, little to no evidence of some criteria was detected such as Leadership (5), Policy (6), Resources (10), Competence (11), Awareness (12), Communication (13), Monitoring and measurement (18), Internal audit (19), Management review (20), Nonconformity and corrective action (21), and Continual improvement (22).

The results in the previous section indicated that far too little attention has been paid to address all or most requirements of the standard. Whilst some research were produced in years in question, but only two studies attempted to investigate the ISMS at a proficient level, which meeting between 10 to 14 criteria out of 22. No study was found to reach the advanced level, meaning to support more than 15 criteria out of 22. The evidence affirms that the majority (57%) of the selected studies are at the developing stage, which means they are only able to fulfil up to four (18%) requirement of the standard.

The average fulfilment rate of all the 21 studies is 23% which is equivalent to five out of 22 requirements. Excluding any of the 22 requirements specified in Table II is not acceptable when an organisation claims conformity to the standard. Taken together, the current studies are incomplete and the current literature requires further expansion. The number of studies at each category stands as below:
Basic = 7 Developing = 12
Proficient = 2 Advanced = 0

This interpretation contrasts with findings in Table III which provide detailed results about the scope of each study at the PDCA model.

A considerable number of studies mostly fulfil the requirements of the Plan (26%) and Do (52%) stages, whilst, very few studies attempted to address Check (19%) and Act (3%) stages. Additionally, it reveals that nearly two-thirds of papers were considered as a Method whereas only seven papers were identified as a Framework. It is also worth noting that a distinct number of other papers which were dismissed during the selection process and not considered for the purpose of this systemic literature review were identified as Method.

A note of caution is due here since not every study selected as part of our review claimed to specifically meet each and every requirement of the standard and it could be argued that those study only focused on the indicated criteria, however, this suggests that there is a rudimentary gap in the knowledge.

V. CONCLUSION

ISO/IEC 27001 standard provides mapping for establishing, implementing, maintaining, and continually improving an information security management system. It is a strategic decision for organisations to adopt ISMS and systematise the security demands rather than doing it ad hoc.

The aim of this paper is to systematically investigate the approaches, which assist organisations to develop, implement, and comply with the ISMS as defined by the standard published between 2005 to 2018. The review examined the approaches, which are thought to contribute to the requirements of the ISO/IEC 27001 standard.

The results of this study indicate that there is no concept that facilitates organisations in designing and complying with the ISO/IEC 27001 standard, this means covering all the 22 requirements of the standard. The result indicates that a very limited number of studies attempted to investigate the ISMS at the proficient level which meets at least half of the requirements of the standard. No study was found to address all parts of the ISMS. The average fulfilment rate of all 21 studies is 23% which is equivalent to addressing up to 5 out of 22 requirements.

The majority of the primary studies were produced prior to the 2013 and the current literature requires attention in expanding, promoting, and the introduction of techniques to organisations and enable them to achieve and strategically integrate their organisational goals with their information security objectives.
The most obvious finding to emerge from this study is that substantial potential exists for academic researchers to investigate the ISMS under a holistic approach. The current literature
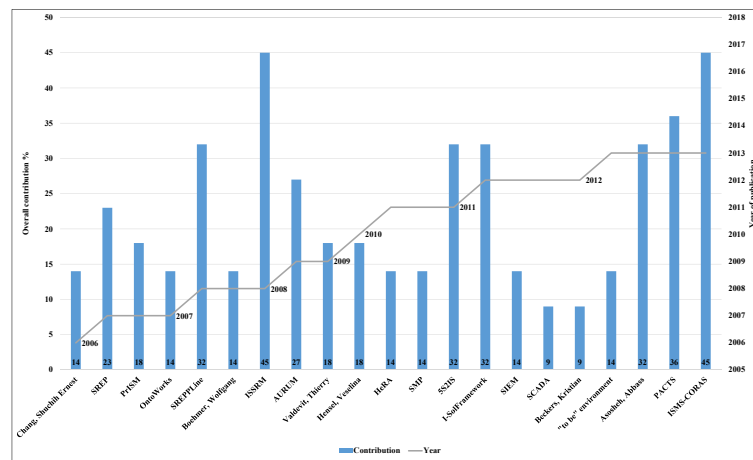
Figure 3. Timeline comparison of each study in (%)

TABLE IV. Detailed view of the studies

Title	Plan														Do			Check			Act		Overall
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	
Chang, Shuchih Ernest	*	*									*												Developing
SREP	*	*						*	*							*							Basic
PrISM																		*	*	*		*	Developing
OntoWorks															*			*	*				Developing
SREPPLine	*	*	*				*	*	*							*							Basic
Boehmer, Wolfgang								*	*						*								Developing
ISSRM	*	*	*				*	*	*						*	*	*			*			Proficient
AURUM	*	*						*	*							*	*						Basic
Valdevit, Thierry			*									*		*									Developing
Hensel, Veselina	*	*														*	*						Developing
HeRA	*	*		*																			Developing
SMP														*				*		*			Developing
5S2IS	*		*		*				*						*			*		*			Basic
I-SolFramework	*	*	*			*		*		*	*												Basic
SIEM															*	*	*						Developing
SCADA															*	*							Developing
Beckers, Kristian														*		*							Developing
"to be" environment								*								*	*						Developing
Asosheh, Abbass	*			*				*	*						*	*	*						Basic
PACTS	*	*		*	*			*						*	*	*							Basic
ISMS-CORAS	*	*	*	*				*	*					*	*	*	*						Proficient

Note:
Developing = Fulfil up to 4 criteria out of 22
Basic = Fulfil between 5 to 9 criteria out of 22
Proficient = Fulfil between 10 to 14 criteria out of 22
Advanced = Fulfil more than 15 criteria out of 22

lacks motivation in purposing new initiatives in the field of the ISMS and ISO/IEC 27001 standard.

In the absence of the knowledge in the literature, there is abundant room for further progress in determining a technique which, could handle the ISMS under a unified approach. The evidence from this review suggests that there is a gap in the current approaches to satisfy the requirements of the ISO standard and fresh and comprehensive approaches are beneficiary and recommended in advancing the ISMS.

Future research should be carried out to examine more closely the links between all four scopes of the PDCA model. More research is needed to develop a deeper understanding of the relationships between ISO/IEC 27001 standard and requirements engineering and greater efforts are required to ensure that future approaches to account for all requirements of the standard.

REFERENCES

[1] D. Ganji, H. Mouratidis, and S. Malekshahi Gheytassi, "Towards a Modelling Language for Managing the Requirements of ISO/IEC 27001 Standard," in 5th International Conference on Advances and Trends in Software Engineering (SOFTENG). Valencia, Spain: IARIA, 2019, pp. 17–23.

[2] J. P. Laudon and K. C. Laudon, Essentials of MIS, global edition, 12th ed. Pearson Higher Education & Professional Group, 2016.

[3] E. Targett, "6 months, 945 data breaches, 4.5 billion records," 2018, (Accessed: 2019-11-01). [Online]. Available: https://www.cbronline.com/news/global-data-breaches-2018

[4] The Breach Level Index, "Data breach database," (Accessed: 2019-11-01). [Online]. Available: https://breachlevelindex.com/data-breach-database

[5] K. C. Laudon and J. P. Laudon, Management information systems: managing the digital firm, 10th ed. Pearson Higher Education & Professional Group, 2007.

[6] International Organization for Standardization, "ISO/IEC 27001

Information security management," (Accessed: 2019-11-01). [Online]. Available: https://www.iso.org/isoiec-27001-information-security.html

[7] International Organisation for Standardisation, "The ISO survey of management system standard certifications 2017," International Organisation for Standardisation, Geneva, Switzerland, Tech. Rep., 2017.

[8] B. Von Solms, "Information Security governance: COBIT or ISO 17799 or both?" Computers and Security, vol. 24, no. 2, 2005, pp. 99–104.

[9] B. A. Kitchenham, "Guidelines for performing systematic literature reviews in software engineering," Keele University, Keele, UK, Tech. Rep., 2007.

[10] ——, "Procedures for performing systematic reviews," Keele University, Keele, UK, Tech. Rep., 2004.

[11] J. Webster and R. T. Watson, "Analyzing the past to prepare for the future: writing a literature review," MIS Quarterly, vol. 26, no. 2, 2002, pp. xiii–xxiii.

[12] "ISO 27001 global report," IT Governance, Tech. Rep., 2016.

[13] E. Coles-Kemp, "The anatomy of an information security management system," Ph.D. dissertation, King's College London, University of London, 2008.

[14] E. W. Bernroider and M. Ivanov, "IT project management control and the Control Objectives for IT and related Technology (CobiT) framework," International Journal of Project Management, vol. 29, no. 3, 2011, pp. 325–336.

[15] J. McGowan, M. Sampson, and C. Lefebvre, "An evidence based checklist for the peer review of electronic search strategies (PRESS EBC)," Evidence Based Library and Information Practice, vol. 5, no. 1, 2010, pp. 149–154.

[16] M. Rouse, "Framework," 2015, (Accessed: 2019-11-01). [Online]. Available: http://whatis.techtarget.com/definition/framework

[17] C. R. Kothari, Research methodology: methods and techniques. New Age International Publishers, 2004.

[18] "ISO/IEC 27001:2013 Information technology - security techniques - information security management systems - requirements," International Organization for Standardization, Geneva, Switzerland, Tech. Rep., 2013.

[19] British Standard Institution, "Moving from ISO/IEC 27001:2005 to ISO/IEC 27001:2013," British Standard Institution, Tech. Rep., 2014.

[20] S. E. Chang and C. B. Ho, "Organizational factors to the effectiveness of implementing information security management," Industrial Management & Data Systems, vol. 106, no. 3, 2006, pp. 345–361.

[21] S. E. Chang and C.-S. Lin, "Exploring organizational culture for information security management," Industrial Management & Data Systems, vol. 107, no. 3, 2007, pp. 438–458.

[22] D. Mellado, E. Fernandez-Medina, and M. Piattini, "A common criteria based security requirements engineering process for the development of secure information systems," Computer Standards & Interfaces, vol. 29, 2007, pp. 244–253.

[23] ——, "A comparison of the common criteria with proposals of information systems security requirements," in 1st International Conference on Availability, Reliability and Security. Vienna, Austria: IEEE, 2006.

[24] M. M. Anwar, M. F. Zafar, and Z. Ahmed, "A proposed preventive information security system," in 2007 International Conference on Electrical Engineering. Lahore, Pakistan: IEEE, 2007, pp. 1–6.

[25] S. Fenz, G. Goluch, A. Ekelhart, and E. Weippl, "Information security fortification by ontological mapping of the ISO/IEC 27001 standard," in Pacific Rim International Symposium on Dependable Computing. Melbourne, Victoria, Australia: IEEE Computer Society, 2007, pp. 381–388.

[26] S. Fenz, "Ontology-based generation of IT-Security metrics," in ACM Symposium on Applied Computing. Sierre, Switzerland: ACM, 2010, pp. 1833–1839.

[27] S. Fenz, S. Plieschnegger, and H. Hobel, "Mapping information security standard ISO 27002 to an ontological structure," Information & Computer Security, vol. 24, no. 5, 2016, pp. 452–473.

[28] D. Mellado, E. Fernandez-Medina, and M. Piattini, "Security requirements variability for software product lines," in 3rd International Conference on Availability, Reliability and Security. Barcelona, Spain: IEEE, 2008, pp. 1413–1420.

[29] ——, "Towards security requirements management for software product lines: a security domain requirements engineering process," Computer Standards & Interfaces, vol. 30, 2008, pp. 361–371.

[30] D. Mellado, H. Mouratidis, and E. Fernandez-Medina, "Secure Tropos framework for software product lines requirements engineering," Computer Standards & Interfaces, vol. 36, no. 4, 2014, pp. 711–722.

[31] W. Boehmer, "Appraisal of the effectiveness and efficiency of an information security management system based on ISO 27001," in 2nd International Conference on Emerging Security Information, Systems and Technologies. Cap Esterel, France: IEEE, 2008, pp. 224–231.

[32] ——, "Cost-benefit trade-off analysis of an ISMS based on ISO 27001," in International Conference on Availability, Reliability and Security. Fukuoka, Japan: IEEE, 2009, pp. 392–399.

[33] N. Mayer, P. Heymans, and R. Matulevicius, "Design of a modelling language for information system security risk management," in 1st international conference on research challenges in information science. Ouarzazate, Morocco: IEEE, 2007, pp. 121–132.

[34] N. Mayer, "Model-based management of information system security risk," Ph.D. dissertation, University of Namur, 2008.

[35] ——, "A cluster approach to security improvement according to ISO/IEC 27001," in 17th European Systems & Software Process Improvement and Innovation. Grenoble, France: Springer-Verlag Berlin Heidelberg, 2010.

[36] A. Ekelhart, S. Fenz, and T. Neubauer, "AURUM: a framework for information security risk management," in 42nd Hawaii International Conference on System Sciences. Big Island, HI, USA: IEEE, 2009, pp. 1–10.

[37] ——, "Ontology-based decision support for information security risk management," in Fourth International Conference on Systems. Gosier, Guadeloupe, France: IEEE, 2009, pp. 80–85.

[38] T. Valdevit, N. Mayer, and B. Barafort, "Tailoring ISO/IEC 27001 for SMEs: a guide to implement an information security management system in small settings," in Software Process Improvement: 16th European Conference, EuroSPI, vol. 42. Alcala (Madrid), Spain: Springer, Berlin, Heidelberg, 2009, pp. 201–212.

[39] T. Valdevit and N. Mayer, "A gap analysis tool for smes targeting ISO/IEC 27001 compliance," in 12th International Conference on Enterprise Information Systems, Funchal, Madeira - Portugal, 2010, pp. 413–416.

[40] V. Hensel and K. Lemke-Rust, "On an integration of an information security management system into an enterprise architecture," in International Workshop on Database and Expert Systems Applications. Bilbao, Spain: IEEE, 2010, pp. 354–358.

[41] C. Braun, F. Wortmann, M. Hafner, and R. Winter, "Method construction - a core approach to organizational engineering," in ACM symposium on Applied computing. Santa Fe, New Mexico: ACM, 2005, pp. 1295–1299.

[42] K. Schneider, E. Knauss, S. H. Houmb, S. Islam, and J. Jurjens, "Enhancing security requirements engineering by organisational learning," Requirements Engineering, vol. 17, no. 1, 2012, pp. 35–56.

[43] I. Müller, J. Han, J.-G. Schneider, and S. Versteeg, "Tackling the loss of control: standards-based conjoint management of security requirements for cloud services," in 4th International Conference on Cloud Computing. Washington, DC, USA: IEEE Computer Society, 2011, pp. 573–581.

[44] ——, "Idea: a reference platform for systematic information security management tool support." in Engineering Secure Software and Systems. Madrid, Spain: Springer-Verlag Berlin Heidelberg, 2011, pp. 256–263.

[45] A. Gillies, "Improving the quality of information security management systems with ISO 27000," The TQM Journal, vol. 23, no. 4, 2011, pp. 367–376.

[46] H. Susanto, M. N. Almunawar, and Y. C. Tuan, "Information security challenge and breaches : novelty approach on measuring ISO 27001 readiness level," International Journal of Engineering and Technology, vol. 2, no. 1, 2012, pp. 67–75.

[47] ——, "A novel method on ISO 27001 reviews: ISMS compliance readiness level measurement," Computer Science Journal, vol. 2, no. 1, 2012, pp. 19–29.

[48] H. Susanto, M. N. Almunawar, Y. C. Tuan, and M. S. Aksoy, "I-Solframework: an integrated solution framework six layers assessment on ultimedia information security architecture policy compliance," International Journal of Electrical & Computer Sciences, vol. 12, no. 01, 2012, pp. 20–28.

[49] R. Montesino, S. Fenz, and W. Baluja, "SIEM-based framework for security controls automation," Information Management & Computer Security, vol. 20, no. 4, 2012, pp. 248–263.

[50] R. Montesino and S. Fenz, "Automation possibilities in information security management," in European Intelligence and Security Informatics Conference. Athens, Greece: IEEE, 2011, pp. 259–262.

[51] M. P. Azuwa, R. Ahmad, S. Sahib, and S. Shamsuddin, "A propose technical security metrics model for SCADA systems," in International Conference on Cyber Security, Cyber Warfare and Digital Forensic. Kuala Lumpur, Malaysia: IEEE, 2012, pp. 70–75.

[52] M. P. Azuwa and R. Ahmad, "Technical security metrics model in compliance with iso/iec 27001 standard," International Journal of Cyber-Security and Digital Forensics (IJCSDF), vol. 1, no. 4, 2012, pp. 280–288.

[53] K. Beckers, S. Faßbender, M. Heisel, and H. Schmidt, "Using security requirements engineering approaches to support ISO 27001 information security management systems development and documentation," in 7th International Conference on Availability, Reliability and Security. Prague, Czech Republic: IEEE, 2012, pp. 242–248.

[54] K. Beckers, S. Faßbender, M. Heisel, J.-C. Kuster, and H. Schmidt, "Supporting the development and documentation of ISO 27001 information security management systems through security requirements engineering approaches," in 4th International Symposium on Engineering Secure Software and Systems. Eindhoven, The Netherlands: Springer, Berlin, Heidelberg, 2012, pp. 14–21.

[55] A. Chatzipoulidis, A. Belidis, and T. Kargidis, "A risk management approach to information society," The University of the Fraser Valley Research Review, vol. 4, no. 3, 2013, pp. 42–56.

[56] A. Asosheh, P. Hajinazari, and H. Khodkari, "A practical implementation of ISMS," in 7th International Conference on e-Commerce in Developing Countries with focus on e-Security, vol. 11. Kish Island, Iran: IEEE, 2013.

[57] K. Beckers, I. Cote, S. Faßbender, M. Heisel, and S. Hofbauer, "A pattern-based method for establishing a cloud-specific information security management system," Requirements Engineering, vol. 18, no. 4, 2013, pp. 343–395.

[58] K. Beckers, M. Heisel, I. Côté, L. Goeke, and S. Güler, "Structured pattern-based security requirements elicitation for clouds," in International Conference on Availability, Reliability and Security. Regensburg, Germany: IEEE, 2013, pp. 465–474.

[59] K. Beckers, S. Faßbender, and M. Heisel, "A meta-model approach to the fundamentals for a pattern language for context elicitation," in 18th European Conference on Pattern Languages of Program. Irsee, Germany: ACM, 2013, p. 28.

[60] K. Beckers, H. Schmidt, J.-C. Kuster, and S. Faßbender, "Pattern-based support for context establishment and asset identification of the ISO 27000 in the field of cloud computing," in 6th International Conference on Availability, Reliability and Security. Vienna, Austria: IEEE Computer Society, 2011, pp. 327–333.

[61] K. Beckers, M. Heisel, B. Solhaug, and K. Stolen, ISMS-CORAS : A structured method for establishing an ISO 27001 compliant information security management system. Springer, Cham, 2014.

[62] K. Beckers, Supporting ISO 27001 establishment with CORAS. Springer, Cham, 2015.

Open Data Based Digital Platform for Regional Growth and Development in Norway: A Heuristic Evaluation of the User Interface

Salah Uddin Ahmed[1], Fisnik Dalipi[2], Steinar Aasnass[1]
[1]University of South-Eastern Norway, Hønefoss, Norway
[2]Linnaeus University, Kalmar, Sweden
E-mail: {salah.ahmed, steinar.aasnass}@usn.no; fisnik.dalipi@lnu.se

Abstract—**Even though a homogenous growth is more desired and expected, socio-economic disparities can still be observed across different regions of a country. Measuring regional growth is essential for the regions that are falling behind or struggling to achieve desired growth. While there exists different statistics of regional growth data, there is a lack of digital tools that represent the growth data in a visual friendly manner to the different stakeholders involved in the region. We have developed a digital platform to address this issue for a region in Norway. The platform is developed to use open data from different sources that is presented in five major groups related to growth: goals, premises or prerequisites for growth, industries, growth, and expectations. The platform helps to improve decision-making and transparency, as well as provide new knowledge for research and society. Compared to other similar digital platforms from Norway, our platform has the strength of providing more options for visualization that makes the statistics more comprehensive. However, we believe that proper usability of the tool can be improved by getting feedback from real users. Therefore, a heuristic evaluation is conducted in order to find out the usability issues and further improve the tool for its users. In this article, we report the results of evaluation and implications for the future improvement along with the presentation of our tool.**

Keywords-digital platform; growth barometer; regional growth; open data; analytics; visualization; usability; heuristic evaluation.

I. INTRODUCTION

This paper is an extended version of our previous work [1], where we presented *Vekstbarometer*, an open data and analytics driven digital platform for regional growth and development used in a municipality in Norway. Open data nowadays represent an indispensable source for governments' strategy when it comes to coping with many innovation challenges of the future. In addition, open innovation philosophies and approaches are continuously being initiated and ratified by public sectors in many different countries [2][3].

In Norway, many governmental agencies have adopted the open data initiative and are making data accessible for general use. Hence, in order for many businesses and even citizens to create innovative value-added products and services, they can now access and utilize these open data resources [4]. Businesses while relying on open data resources can make more intelligent and effective moves by performing data analytics on top of open data.

When defining the meaning of open data, we use the definition provided by Open Data Institute, which defines open data as "data that is made available by government, business and individuals for anyone to access, use and share" [5]. It is worth noting that the global economic potential value of open data has been estimated to $3 trillion [6]. However, the potential and advantage of Open Government Data (OGD) for improving services in different economic sectors has not been realized to a large extent [7].

Vekstbarometer is an application that documents and illustrates our effort of using open data provided by *Statistics Norway* [8], *Real Estate Norway* [9] and the Brønnøysund *Register Centre* [10], to create an analytics-driven web application that presents regional indicators together with research-based knowledge relevant and applicable to regions' growth. As illustrated with the conceptual model shown in Figure 1, regional growth correlates with growth in Value Creation, Employment, Workplace and Population. A region's growth is often measured by growth in GDP (Gross Domestic Product). Nonetheless, the main goal of the policy is to contribute to higher welfare and transform the region into better place to live and run business. In order for the growth barometer to be able to clarify developments and provide compatible information connected to political and business decisions, a comprehensive definition of growth is therefore necessary. The indicators are based on open data while statistical visualizations can be created also for purposes other than regional growth. In this study, we have focused in the region of Ringerike, located in the southeastern part of Norway, which has recently seen a decline in the number of jobs and insufficient economic growth. To the best of our knowledge, there is no similar system developed so far, which includes research-based knowledge on a regional level in Norway.

In a national level, one can encounter various digital platforms, but they are all different in the sense that they are using different data categories, but also, they have different purposes. Above all, they provide a limited set of key indicators, which visualize, and, are associated with regional growth and development.

A considerable number of regions have business barometers based on survey data, gathered on yearly basis and register data. They can predict the national and regional

business trends. One such example is Konjukturbarometer Østlandet [11], a digital platform that contains, inter alia, a knowledge-based database covering information on developments in the counties of Hedmark, Oppland, Oslo, and Akershus. The Confederation of Norwegian Enterprise (NHO), which is Norway's largest organization for employers and the leading business lobbyist, represent another case of digital platform. Their platform, Økonomibarometeret [12] encompasses the current market situation, operating profit, investments, and employment on a county and national level.

The main objective behind developing our innovative system, which can be accessed at *vekstbarometer.usn.no* or with the path *vekstbarometer.usn.no/ringerike,* is to offer public sector organizations and local businesses a management tool with key indicators related to the region's growth. Besides having the feature of improving public sectors' transparency and engagement of civil society, our system can also add value towards enhancing economic growth via processing, interpreting, and illustrating regional open data in a comprehensive and meaningful way.

The remainder of this paper is structured as follows: Section II presents related work while Section III informs the research context: Ringerike region's need of a digital platform for its development. Section IV introduces the *Vekstbarometer* system and the technology used for its development, along with its strength and impact for regional growth. Section V presents the heuristic evaluation of the user interface of the application and lastly, Section VI concludes the paper and provides some insights about potential future work.

II. RELATED WORK

As a result of the emergence and pervasiveness of ICT, many governmental institutions across the globe have been embracing initiatives to transform themselves into e-governments [13], and consequently are encouraging citizen participation in governments. OGD represents one of the main extensions of such e-government initiatives [14]. OGD is making data freely available to all with the sole intention to ensure public accountability and transparency [15], to boost innovation in various economic sectors and to increase efficiency in administration.

But, notwithstanding this, for the stakeholders to derive the public value out of the open data, it is of paramount importance for the data sets to be re-usable, comprehensive, interpretable, complete, and permit user-friendly interface. Also, government authorities should be proactive to ensure that the data sets are in harmony with the stipulated norms, such as preserving personal and private information of the users, or protecting and prohibiting the publishing of sensitive data associated to national security. In that regard, the government has created the *Norwegian License for Open Government Data* (NLOD) and have recommended all data owners in the public sector to apply the license, which contains, among others, information on preserving confidential and personal data [16]. When it comes to global status and trends of open data in the context of readiness, implementation and impact, Norway ranks among the top ten best countries in the world [17].

In the state-of-the-art literature, there exist two categories of OGD research. The first category is mainly based on conceptual models and frameworks including theoretical proposals [18, 19], also incorporating studies, which are discussing the main stages of the OGD life cycle [20]. The second category, where our work belongs, incorporates studies that are taking place in different geographical locations by using OGD at the state or local level. Additionally, one important characteristic of this category is the open data exploration and exploitation. Here, the data generated by multiple sources of governmental institutions are processed and visualized for different purposes. This includes conducting various analysis, creating mashups, to improve the interpretability of open data, or even innovate upon the open data.

Recently, multiple applications have been successfully implemented and developed based upon the open data sets across the world, focusing primarily on larger cities, such as Chicago [21][22], New York [23], Dublin [24], St. Petersburg [25], Singapore [26].

However, despite the interest and the rapid proliferation of open data platforms, many challenges remain when it comes to the accessibility and usability of platforms using open data, data quality and completeness, and interpretability of open data. As far as enhancing the interpretability of open data is concerned, the authors in [21][22] run a case study to analyze the open socioeconomic data released by the city of Chicago, where they use different visualizations adjusted for univariate, bivariate and multivariate analysis. This approach implies that delving into open urban data can lead to more effective data interpretation and analysis. Regarding the usability aspects for user interface design of open data platforms, a case study scenario also including a transportation challenge in Dublin City identified important patterns for highly usable open data platforms for open data policy [24]. This study recommends that these platforms should adopt user-friendly technology and social media platforms. From architectural standpoint, in the literature there is an interesting and relevant work presented by [26]. In this work, an open data platform prototype is developed to illustrate the requirements and the architecture of open real time digital platform. The main aim behind developing this platform is to serve as a base for programming the city of Singapore, and generate visual data analytics in a city context.

Nonetheless, none of the mentioned research works was related to comprehensive visualization of growth data fetched from open governmental datasets in a regional context. Therefore, we aim to fill this gap by developing *Vekstbarometer,* an open data driven digital platform which will aid businesses in making more intelligent and strategic moves. The platform will also help businesses to perform

informed decision making and will increase transparency for citizens.

III. RESEARCH CONTEXT: REGION'S NEED FOR A DIGITAL PLATFORM

The *Vekstbarometer* application was developed for Ringerike region as a response to deal with the regions falling growth. Many key factors related to growth and welfare pointed out lack of desired growth in the region.

Ringerike has around 43,000 inhabitants. Over the last 10 years, as mentioned in [27], the number of total new employment for the region was only 145 while at the same time the number of jobs decreased was 321. Besides, value creation for the region's business sector has also shown a weaker development compared to other neighboring regions that are natural competitors for Ringerike having, similar background, assets and distance from capital city. If this negative tendency is not reversed in near future, Ringerike region will face the consequences of declining private and public welfare. This will make the region a less attractive place to live. However, politicians and businesses in the region are optimistic in terms of the future. New four-lane highway and new high-speed railway from the capital Oslo to Ringerike region's main city Hønefoss is planned to be completed around 2028. This gives enormous opportunities for Ringerike to reverse the negative trend and generate new growth that will ensure future welfare and good living conditions in the region. Nevertheless, in order to accomplish this growth, it requires good decisions from the region's public authorities and industry. A growth barometer that monitors the growth and significant conditions for growth in the region could provide a useful management tool for strategic decisions.

The target user group for the barometer are politicians, municipality administration, business and the community.

Large amounts of statistics and information related to growth and development for municipality regions already exists from different sources. However, there is a need that these statistics and information are provided in a customized way targeted for the use in Ringerike region besides developing new and better statistics for the region, and breaking them down at regional and local level. In our search of finding similar tools or platforms as growth barometer for other regions in Norway, we found quite a few solutions, but none with the approach and objectives that we mentioned here. We believe that a digital platform like growth barometer could give Ringerike region a significant competitive advantage over other regions, which are also seeking regional growth and development.

A. The logical model of Barometer

The conceptual model behind the growth barometer is given in Figure 1. The objective of regional development is given through the points in (A), i.e., higher value creation, employment, jobs and higher populations. In the national level, there will always be a number of people and businesses considering to "relocate". A region will compete with other regions as a national competition to get these businesses and individuals to establish themselves in its own territory. Here, conditions for regional growth (B) and local industry's contribution to local growth (C) can serve as lucrative points for capturing a good share of the influx of national movements of persons and businesses. If the conditions (B) and the local business sector's efforts (C) are large, this portion will be large. The result will be a major influx of businesses and individuals. We assume that growth targets mentioned in (A) can be achieved together with regional economic dynamics linked to business development, housing construction, etc.
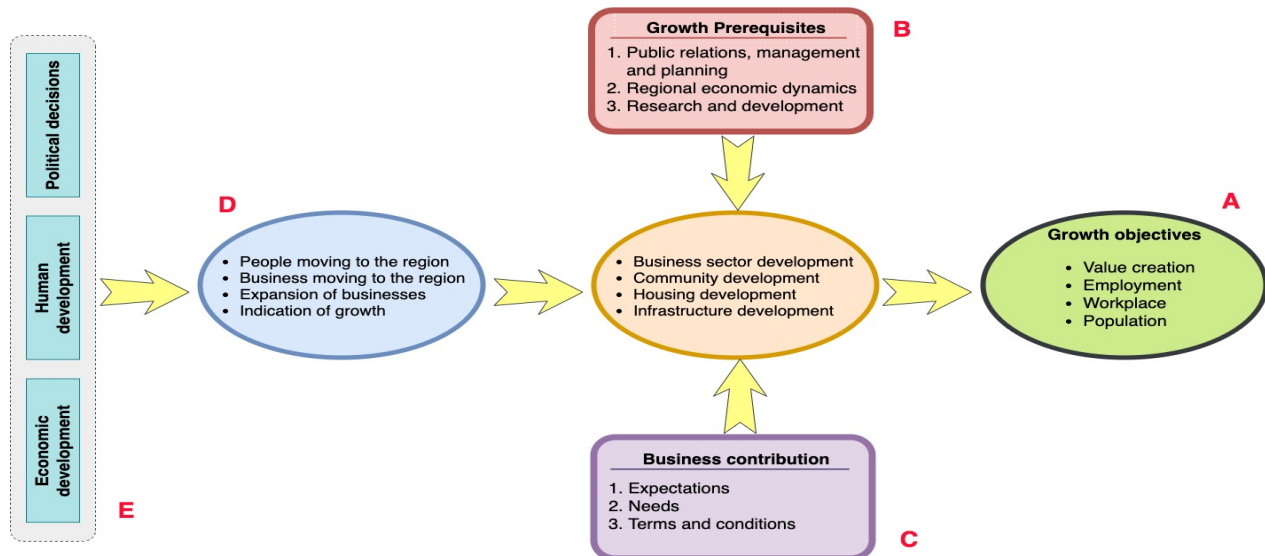
Figure 1. The logical model behind the *Vekstbarometer*

IV. VEKSTBAROMETER - THE DIGITAL PLATFORM TECHNOLOGY

Vekstbarometer, which in Norwegian stands for "growth barometer" is a digital platform that provides the development trends in the regional context in a visual and user-friendly way. The platform uses data from different sources, these data are presented mainly in five main groups: i) Goals, ii) Premises or Prerequisites for growth, iii) Industries, iv) Growth, and v) Expectations.

Each group consists of several categories and each category contains several variables, which actually holds the statistical data. The groups and the categories form the information architecture of the digital platform, which is shown in the Figure 2.

The group *Goals* contains the following categories: *Population, Value Creation, Employment, Jobs,* and *Welfare*. Each of these categories contains a number of variables, which for the sake of simplicity, are not shown in the figure. These variables represent the data analyses and are used to generate the visual representations in the form of charts, graphs, and diagrams. For example, the category *Population* contains ten variables, some of which are: total inhabitants, age-wise population, population change trends, net population change etc. Each variable is represented by a number, for example total inhabitants is presented by 1, age-wise population is presented by 25. The numbers are not assigned in any particular order; rather they were assigned when the statistics of the variables were being added in the system.

The variable number can be seen from the URL; when browsing a certain statistic from the navigation menu the URL gets changed with the variable number such as https://vekstbarometer.usn.no/statistic/25.
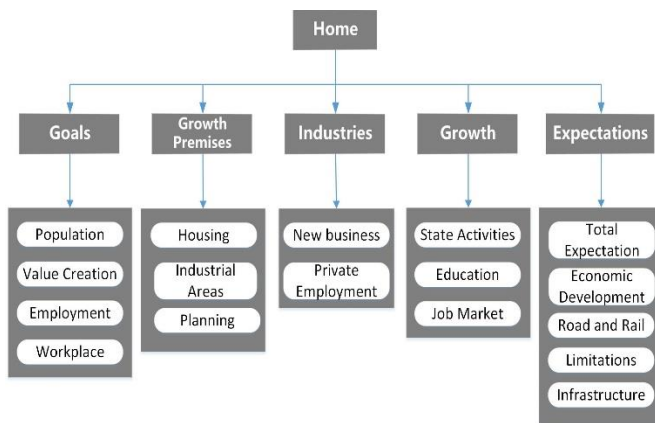
Figure 2. Information architecture of the application

Thus, the statistic variable 25 can be accessed from the navigation menu, as well as from the URL by appending the variable number (25) after the URL given at [28].

In addition to presenting open data, Vekstbarometer also presents survey data from the local industries that reflects the expectations and assumptions of the local entrepreneurs and business owners. The application consists of mainly three parts: backend, presentation and database. The system architecture of *Vekstbarometer* application is given in Figure 3.

Backend is mainly responsible for processing requests from the presentation layer. Based on the requests, it retrieves raw data from the database, sort them and send them to the presentation layer. Backend collects data from external data storage such as open data that are provided by others (such as Statistics Norway) through APIs. It also processes other input data to the system and the survey data. Data from external sources are retrieved in the desired formats and are saved in the database for storage. Different external data are updated in different periodic intervals, such as quarterly, half-yearly and yearly. These data sources are fetched periodically to keep the main database updated.
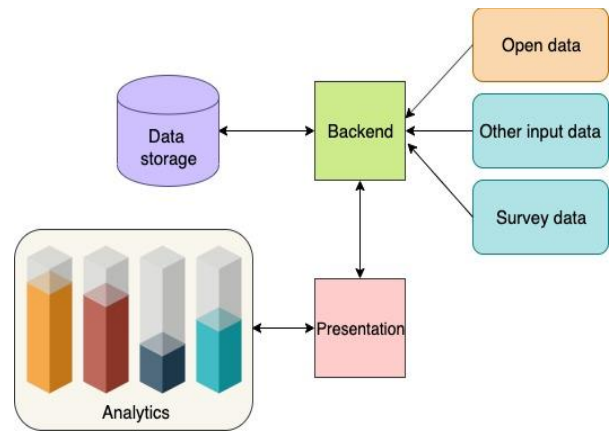
Figure 3. *Vekstbarometer* system architecture

Presentation takes care of conversion of data transmitted from backend, transformation of these data and displaying on-screen to end-user. This is the main part of the application that the user sees and interacts with.

The database holds the persistent and transient data that are critical for running the application. At the beginning, we have used *MySQL* databases for the persistent storage, which was later migrated to MariaDB. We have used *Angular 2* for presentation and frontend, and *PHP* for the backend.

While developing the digital platform, at first the relevant growth indicators and variables were carefully selected and matched with the region's context. Relevant indicators were chosen based on our previous research and then they were matched with regional contexts through several meetings with local stakeholders, key business people and municipality administrators. The relevant regions to which Ringerike's statistics are compared with were also selected through these meetings.

Later on, based on the variables, suitable data providers were searched and identified for them. In case of selecting a data provider, the accuracy and trustworthiness were prioritized. When multiple providers were found for a single

variable, the originality or the owner of the data was sought. In order to avoid any discrepancy, we followed the principle of collecting data from the original providers and in case of multiple sources of some data; we collected all similar data from a single source to have validity of comparisons among them.

Most of the data used in the platform is collected from *Statistics Norway,* which provides data from country level to municipality level for all municipalities in Norway. It provides an API service to query through its 6000 tables and output is provided in *JSON-stat*, *csv*, and *.xlsx* formats.

For the initial phase of the application, we collected the desired data for one or more municipalities of the barometer by using Statistics Norway's API console, and stored them in the application's database. Subsequently, an admin panel was added in the application from where admin can update these tables whenever new data is available in the source providers.

A. Data

The growth barometer application *Vekstbarometer* is a data visualization platform. It deals with several kinds of data some of which are sensitive data and some are not sensitive data. Data that are already presented as open data by other sources, such as *Statistics Norway* are non-sensitive. Since these data are already available and open for public, we do not have any restrictions on showing them in raw format or in any modified format. Apart from non-sensitive data, there are data or part of data which contain some sensitive information about people or businesses that should not be made publicly available, e.g., a person's personal number (social security number) or a business's sales and marketing plans, confidential customer or supplier information, etc.

Vekstbarometer also stores and shows data collected through surveys. These surveys are done anonymously and therefore personal information and anonymity should be handled as sensitive. Besides, we follow the EU General Data Protection Regulation (GDPR) rules when collecting data from local businesses.

Data that are collected from other sources are mostly obtained by running some data processing operations such as filtering or making queries in order to fit our local needs that are represented by the statistics variables. Referring to the Article 4 EU GDPR definitions, we can say that data processing is a broad term that includes several operations such as collection, recording, organization, structuring, storage, adaptation or alteration, retrieval, consultation, use, disclosure by transmission, dissemination or otherwise making available, alignment or combination, restriction, erasure or destruction etc. [29]. Being a data driven platform, *Vekstbarometer* includes lot of these data processing activities as the data provided by the sources come in various formats, various shapes and they require to be adapted to a standard format before being stored in the application's database.

Vekstbarometer, though mostly presents selected variable/indicator's data from other sources in suitable visual friendly graphical presentations, it also provides the users some pre-selected operations such as selecting and deselecting range of parameters and regions for a certain statistic variable. Besides, the growth barometer has the capability to show some newly created data that are not available in any other data sources.

B. Visualizations

Data in the growth barometer is presented using different kinds of charts and diagrams which include line charts, bar charts, stacked columns, stacked percentage column, column with drill down, pie charts etc. For visualizations, we have used a JavaScript library called *Highcharts*. With *Highcharts*, it is possible to create charts in many shapes, like heat maps, waterfall series and more. In addition, the charts are highly configurable, customizable and interactive.
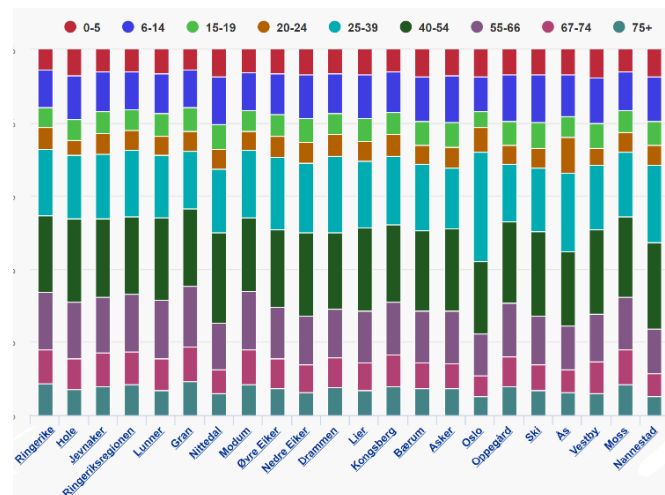
Figure 4. Age wise population chart for all regions, variable 25

When we say configurable and customizable, we mean that values can be added and removed to the charts on the fly. As an example, if we take a look at Figure 4, which presents the graph for variable 25 and shows the age-wise population comparisons in different regions; we can choose the age groups that will be presented in the graph. The graph shows not only the percentage of an age group, in addition, it also the absolute value of that group in a tooltip text when mouse is hovered over on that group. Besides these customization features, the regions in the x-axis is linked to a detailed view of the age wise population chart for only that region (Figure 5), thus making the visualization configurable and interactive at the same time.
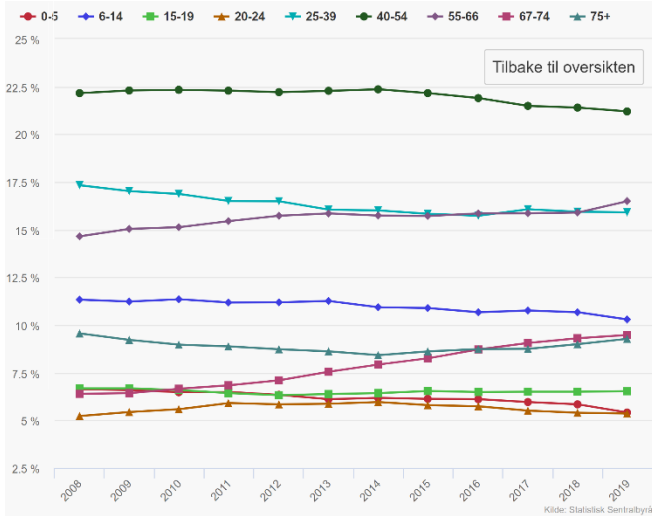
Figure 5. Age wise population chart for a specific region, variable 25

The visualization of a statistic variable is further complemented by some additional information such as: *i) textual description of the variable*, *ii) links to related documents iii) links to related variables*. Besides, each variable presentation is provided with *alternative graph/presentation options*. Accompanying textual description gives a brief introduction of the variable for a better understanding of the variable and its usage. Users can read the *links to related documents* for further understanding of the concepts. By using the *links to related variables*, users can navigate to related statistic variables directly from a specific variable's page. The *Alternative graph option* lists a number of options, and by selecting one of those options, the user can view the same statistical variable in a different form of presentation, for example in linear chart, bar chart, pie chart etc., which could give another view and perspective of the variable.

C. The Strength and Impact of the platform

As reported in our previous study [1], the growth barometer digital platform shows data in an easy, comprehensible and meaningful way. Although some data that our platform presents are already provided by other data outlets, our platform adds extra value by making a better presentation of the data. Furthermore, in contrast to other platforms, such as [30][31][32][33][34], where the data is presented as raw data or with basic level of presentations, our platform has the advantage of providing a range of options for visualization that makes the statistics more comprehensive. Even in case where the platform is showing existing data from other sources, still we add value on it.

In addition to showing already existing data, the platform shows some new data as well. The new data can be of two types: i) newly formed data, ii) newly created data. Newly formed data is created by combining and filtering data from multiple sources where the data were partly available but not

in the exact form that is presented here in the growth barometer.

On the other hand, newly created data refers to the case where we create, collect or gather new data that were not available or presented in any other platform. As an example of newly formed data, we can mention "future population growth", which is represented in our platform as statistical variable 56 (see Figure 6). Here, the expected *future population growth* is taken from three different sources: *Statistics Norway* estimation, political estimation and housing building estimation. This new variable shows the gap or surplus of housing capacities with the *Statistics Norway* estimation or political assumptions and gives a good indication whether the assumptions or planning are feasible or not compared to the housing capacities that are planned for the region. In addition to showing the statistics in visual form, the platform also provides the option to download data and images. It has the options to download data in multiple formats such *as csv, pdf, xls*, and the images in *png, jpeg* and *svg vector* image formats. It also provides the option of printing the chart and view the data of the chart in tabular forms. All these options increase the usability of the tool and facilitates multiple use cases for the users of the system. Users of *Vekstbarometer* can use customized graphs and charts from the platform and include them in their presentations or documents.
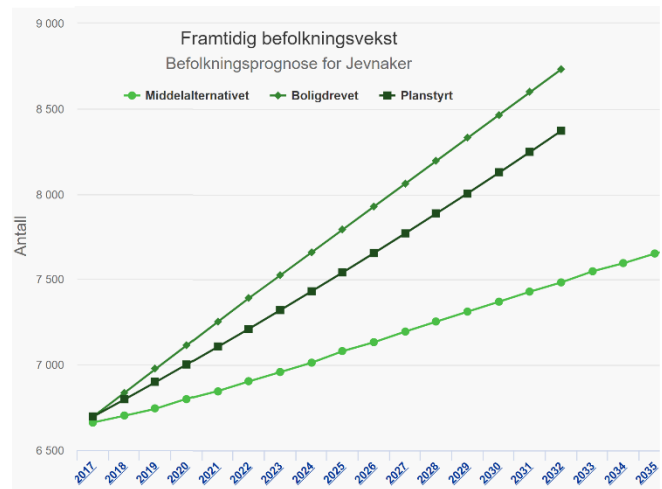
Figure 6. Future population growth, variable 56

The *Vekstbarometer* digital platform is a constituent part of the five years long *Vekstbarometer Project*, which is managed by the *Regional Development Group* at the University of South-Eastern Norway (usn.no). Since the initiation of the project, a status report is planned to be issued every year presenting the current growth status of the region. Ringerike region also has a business policy strategy which sets the premises for how business policy is to be pursued to create growth in the region's business community. The *Vekstbarometer* platform helps to create the strategy and present the annual results of the regional growth according to the growth objectives given at Figure 1. Moreover, the

municipality authorities can rely on *Vekstbarometer* data in order to define their targets and priorities and make better decisions.

V. HEURISTIC EVALUATION

In order to perform the heuristic evaluation of the *Vekstbarometer*'s interface, we have followed the methodology proposed by Jacob Nielsen and Robert L. Mack [35], by having each evaluator inspect the interface independently. As it is recommended by Nielsen on this seminal work, for performing a heuristic evaluation, three to five evaluators are sufficient. In our experiment, four users are involved to examine and evaluate the interface and assess its compliance with recognized usability principles described in [35].

A. Procedure and Data Collection

The evaluators were contacted by phone and were sent information about the survey as well as the questionnaire. Four people were asked and all agreed to participate. The respondents were selected on the basis of their central role in the regional development and their perceived benefit of this application. One of the authors of this paper has also had a dialogue with these people and have understood they have used and were familiar with *Vekstbarometer* web application. The users have been instructed about how the interfaces work and were informed for the purpose of the app, i.e., the evaluators were provided with hints on using the interface. All the evaluators were asked to evaluate *Vekstbarometer* independently. They were very informative and had a lot of relevant feedback as well as the rating of the various criteria in the Nielsen checklist. Duration of the survey including information and some "short discussions" was between 30 and 40 minutes. When it comes to various assessment criteria of the application interface, users were asked to assess the severity of any errors based on the scale given in Table II. In Table I, we present the results from the Nielsen's heuristic evaluation framework for *Vekstbarometer*.

B. Findings and Discussion

Generally, the results from the user evaluation of *Vekstbarometer* indicate positive attitude and experience from our evaluators. All the four evaluators unanimously agree that out of 23 heuristic sub-principles in the checklist, there is no usability problem at all (scale 0) for 1a, 2a, 4a, 5a, 7c, 7d, and 9a, 9b, 9c, which corresponds to the following sub-principles in Table I:

- Visibility of system status/The app provides feedback about status (1a),
- Match between system and real world/Clear terminology/no jargon (2a),
- Consistency and standards/Links are clear and follow conventions (4a),
- Error prevention/Error message problem (5a),
- Flexibility and efficiency of use/Consistent way to return to Home Menu (7c),

- Flexibility and efficiency of use/Simple navigation menu (7d).

The overall evaluation results presented in Figure 7 show that the majority of experienced usability problems by all evaluators correlate with heuristic principles 3 (user control and freedom), 6 (recognition rather than recall), 8 (aesthetic and minimalist design), and 10 (Help and documentation). As far as the first evaluator feedback is concerned (E1), there is no usability issue (scale 0) identified in 19 sub-principles i.e., 1a, 2a, 2b, 3a, 4a, 4b, 4c, 5a, 6b, 7a, 7b, 7c, 7d, 8a, 8b, 8d, 9a, 9b, and 9c from Table I. He identified no cosmetic problems that need to be fixed (scale 1). Three minor usability problems with low priority to be fixed are identified (scale 2), which correspond to principles 3a, 6a, 6c, and only one major usability problem with high priority to be fixed (scale 3) corresponding to 10a.

TABLE I. HEURISTICS EVALUATION RESULTS FOR VEKSTBAROMETER

	Heuristic evaluation checklist	Evaluators			
		E1	E2	E3	E4
1	**Visibility of system status**				
a	The app provides feedback about system status	0	0	0	0
2	**Match between system and real world**				
a	Clear terminology, no jargon	0	0	0	0
b	Content is clear and follow conventions	0	0	2	0
3	**User control and freedom**				
a	Logical structure of the app	0	0	2	2
b	Effective internal search	2	1	3	0
4	**Consistency and standards**				
a	Links are clear and follow conventions	0	0	0	0
b	Various app functions are well integrated	0	0	2	0
c	Clear and consistent navigation throughout the app	0	0	3	0
5	**Error prevention**				
a	Error messages problem	0	0	0	0
6	**Recognition rather than recall**				
a	Objects, actions and options are visible	2	1	3	2
b	Organization of information makes sense	0	1	2	0
7	**Flexibility and efficiency of use**				
a	Very easy to use and interact	0	0	2	0
b	Provides shortcuts for frequent tasks	0	0	3	0
c	Consistent way to return Home/Menu	0	0	0	0
d	Easy to identify current location	0	0	0	0
8	**Aesthetic and minimalist design**				
a	Clean and simple design	0	0	2	0
b	Text and colors are consistent	0	0	2	0
c	Images are meaningful and serve a purpose	2	0	0	0
d	Simple navigation menu	0	0	3	0
9	**Help users recognize, diagnose and recover from errors**				
a	(Error) Messages are explained in a plain language	0	0	0	0
b	Any problem is precisely indicated	0	0	0	0
c	A potential solution is constructively suggested	0	0	0	0
10	**Help and documentation**				
a	Help or Tips are available and clear	3	0	3	3

Scale	Meaning
0	Not a usability problem at all
1	Cosmetic problem only, need not to be fixed
2	Minor usability problem, low priority to be fixed
3	Major usability problem, high priority to be fixed
4	Usability catastrophe, imperative to fix this before product can be released

When it comes to the second evaluator's (E2) results, no usability problems at all (scale 0) were identified in 20 sub-principles, i.e., 1a, 2a, 2b, 3a, 4a, 4b, 4c, 5a, 7a, 7b, 7c, 7d, 8a, 8b, 8c, 8d, 9a, 9b, 9c and 10a.

This evaluator was the only one to discover cosmetic problems (scale 1) relating to principles 3b, 6a, and 6b. No minor usability issues (scale 2) and no major usability problem with high priority that need to be fixed (scale 3) were identified in the rest of 20 heuristic sub-principles.

On the other hand, the third evaluator's (E3) results seem to be more criticizing from the rest of the participants, when it comes to heuristic evaluation of *Vekstbarometer* interface. The feedback from E3 indicate that there are no usability problems at all (scale 0) in 10 out of 23 principles, i.e., 1a, 2a, 4a, 5a, 7c, 7d, 8c, 9a, 9b, and 9c.
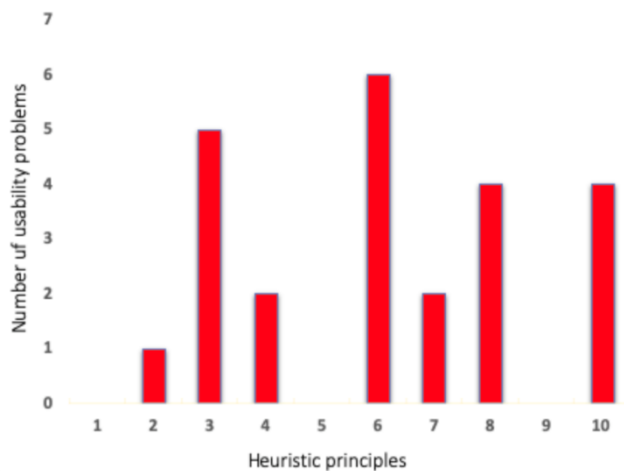
Figure 7. Distribution of usability problems per heuristic principle

Whereas no cosmetic usability problems were identified, still there are 7 minor usability problems (scale 2) discovered in 2b, 3a, 4b, 6b, 7a, 8a, and 8b. Furthermore, this evaluator also identifies 6 major usability problems with high priority of fixing (scale 3), which are corresponding to sub-principles 3b, 4c, 6a, 7b, 8d, and 10a.

Lastly, regarding the fourth evaluator (E4) feedback, only three heuristic sub-principles received usability problem related points. No usability issues at all (scale 0) and no cosmetic problems that need to be fixed (scale 1) were identified in the rest of the 20 heuristic sub-principles. Two minor usability problems with low priority of fixing (scale 2) were identified in sub-principles 3a and 6a, and only one

major usability problem with high priority of fixing (scale 3) was identified for 10a.

As can be seen from Table III, none of the feedback from the four evaluators relates to severity scaling 4 of the Nielsen's heuristic principles, which means that no usability catastrophe is found with this prototype of *Vekstbarometer*. On the other hand, there were altogether 23 usability problems experienced by the 4 participants in heuristics evaluation, of which 3 were cosmetic, 12 were minor, and 8 were major usability problems.

TABLE III.   FREQUENCY OF A SEVERITY SCALE USED

Severity scale	E1	E2	E3	E4	Total
0	19	20	10	20	69
1	0	3	0	0	3
2	3	0	7	2	12
3	1	0	6	1	8
4	0	0	0	0	0

C. Comments from evaluators

In this section, we present some of the comments of the evaluators for the readers. All of the four respondents pointed out that the solution contained highly relevant and useful information relevant to their job. However, all of them mentioned that they had to spend a lot of time getting to know the solution and its functionalities.

"We regularly use Vekstbarometer at meetings. Here we get information that helps us to better understand the connection between different goals set by the municipality." Evaluator #3.

"By using Vekstbarometer we have gained a greater understanding of the factors that are important for the municipality for transforming it into a better place to live." Evaluator #2.

"The Vekstbarometer helps us to see the market potential of our products by studying indicators that describe the business structure of the region." Evaluator #1

"Still many variables of Vekstbarometer application are not fully explained and therefore they make it somewhat unclear." Evaluator #4.

The users were also aware that the solution needed improvement related to help and tips related to navigating in the solution. Since users were familiar with the solution, they found the information they wanted, but were aware that for new users, improved help and tip functionality would be required.

"The search function is bad. It is important and needs to be improved. However, in the meantime, I have become familiar with the content and the functionalities of the

application, so I can manage to find the stuffs that I want." Evaluator #3

"Since the content of the Vekstbarometer is good, I had great motivation to find relevant information from the system despite some weak functionalities." Evaluator #3

"After having a detailed explanation of the functionalities, I see that I have not been able to find all the relevant information. This means that the functionalities needs to be improved." Evaluator #4

The four respondents were respectively a business manager (evaluator 1), an elected representative/politician (evaluator 2), a municipal employee (evaluator 3), and a private consultant (evaluator 4). In overall, they represented well the target group of the *Vekstbarometer*.

VI. CONCLUSION AND FUTURE WORK

In this paper, we have presented the *Vekstbarometer* digital platform and the heuristic evaluation of its user interface. Inspired by the developments in the field of open data initiatives, we developed the platform to combine multiple open data sources to generate various visualizations. This gives insights into the regional growth and development, and demonstrates the usefulness of open data in regional context. Furthermore, it also helps to improve decision-making and transparency, as well as provide new knowledge for research and society. The platform was well accepted by the regional stakeholders. For the purpose of successfully running our experiment, we involved four active users of *Vekstbarometer* to conduct a heuristic evaluation of the tool. The evaluation indicated that the tool has mostly high usability features with a minimal set of major usability problems. Even though the usability problem issues are minimal and mostly coming from one user, still this is a good indication for our future improvement. The application is relatively complex and deals with a lot of statistical parameters and indicators of which many are very similar or has only slight differences among them. Therefore, it is very important to have a well-organized content structure and easy to follow navigation. The evaluation also revealed some of our drawbacks, such as lack of effective internal search, which will help us to focus on our future work.

We also take the positive findings of the heuristic's evaluation as an indication that the *Vekstbarometer* application has been able to provide desired information and interactions to its users and has a good potential for the regional stakeholders.

As a future work, we intend to expand the functionality of the platform by expanding the dataset and focus on converting the digital platform into a fully-fledged and mobile-friendly application. Besides, as a takeaway from the evaluation, we would like to further improve the content structure of the application and implement the internal search options.

REFERENCES

[1] S.U. Ahmed, S. Aasnass, F. Dalipi, and K. Hesten, "Analytics-Driven Digital Platform for Regional Growth and Development: A Case Study from Norway". In Proceedings of The Thirteenth International Conference on Digital Society and eGovernments, Athens, Greece, 2019, pp. 56 – 62.

[2] T. Nam, "Suggesting frameworks of citizen-sourcing via Government 2.0", Government Information Quarterly, vol. 29, pp. 12-20, 2012.

[3] S.M. Lee, T. Hwang, and D. Choi, "Open innovation in the public sector of leading countries", Management Decision, vol. 50, pp. 147-162, 2012.

[4] "Norwegian Directorate for Management and ICT", URL: https://www.difi.no/fagomrader-og-tjenester/digitalisering-og-samordning/apne-data [accessed 2019.12.07].

[5] "Open Data Roadmap for the UK", 2015, The Open Data Institute, URL: theodi.org/article/open-data-roadmap-for-the-uk-2015/ [accessed 2019.12.07].

[6] J. Manyika, M. Chui, P. Groves, D. Farrell, S.V. Kuiken, and E.A. Doshi, "Open data: Unlocking innovation and performance with liquid information". Technical report, McKinsey & Company, 2013.

[7] M. Janssen, Y. Charalabidis, and A. Zuiderwijk, "Benefits, adoption barriers and myths of open data and open government". Information Systems Management, vol. 29, no. 4, 2012, pp. 258–268.

[8] "Statistical Bureauf of Norway", URL:https://www.ssb.no/ [accessed 2019.12.08].

[9] "The Norwegian Housing Price Statistics", URL: http://eiendomnorge.no/ [accessed 2019.12.08].

[10] "Brønnøysund Register Centre", URL: https://www.brreg.no/ [accessed 2019.12.08].

[11] "The Business Barometer *Konjuktur"*, URL: https://konjunkturbarometer1.no/ [accessed 2019.12.08].

[12] "Barometer of the Confederation of Norwegian Enterprises", URL: http://okonomibarometer.nho.no/ [accessed 2019.12.07]

[13] M. Mpinganjira, "Use of e-government services: the role of trust", International Journal of Emerging Markets, vol. 10 no. 4, 2015, pp. 622-633.

[14] T. Jetzek, M. Avital, and N. Bjørn-Andersen, "Generating sustainable value from open data in a sharing society", in B. Bergvall-Kåreborn, and P. Nielsen, (Eds), Creating Value for All Through IT. TDIT 2014. IFIP Advances in Information and Communication Technology, vol. 429. Springer, Berlin Heidelberg.

[15] R.E. Sieber and P.A. Johnson, "Civic open data at a crossroads: dominant models and current challenges", Government Information Quarterly, vol. 32 no. 3, 2015, pp. 308-315.

[16] "Norwegian License for Open Government Data (NLOD 1.0)", URL: https://data.norge.no/nlod/en/1.0 [accessed 2019.12.08].

[17] E. F. Ramos, "Open Data Development of Countries: Global Status and Trends", ITU Kaleidoscope: Challenges for a Data-Driven Society (ITU K), 27-29 Nov 2017, Nanjing, China. DOI: 10.23919/ITU-WT.2017.8246984.

[18] B.W. Wirtz and S. Birkmeyer, "Open government: origin, development, and conceptual perspectives", International Journal of Public Administration, vol. 38, no. 5, 2015, pp. 381-396.

[19] P. McDermott, "Building open government", Government Information Quarterly, vol. 27, no. 4, 2010, pp. 401-413.

[20] J. Attard, F. Orlandi, S. Scerri, and S. Auer, "A systematic review of open government data initiatives", Government Information Quarterly, vol. 32 no. 4, 2015, pp. 399-418.

[21] R. Barcellos, J. Viterbo, L. Miranda, F. Bernardini, C. Maciel and D. Trevisan, "Transparency in practice: using visualization to enhance the interpretability of open data". In Proceedings of The 18th Annual International Conference on Digital Government Research, Staten Island, NY, USA, June 07-09, 2017.

[22] M. Kassen, "A promising phenomenon of open data: a case study of the Chicago open data project", Government Information Quarterly, vol. 30, no. 4, 2013, pp. 508-513.

[23] M. Sutherland and M. Cook, "Data-Driven Smart cities: A closer look at organizational, technical and data complexities". In Proceedings of The 18th Annual International Conference on Digital Government Research, Staten Island, NY, USA, June 07-09, 2017.

[24] E. Osagie, M. Waqar, S. Adebayo, A. Stasiewicz, L. Porwol, and A. Ojo, "Usability Evaluation of an Open Data Platform". In Proceedings of 8th Annual International Conference on Digital Government Research, Staten Island, NY, USA, June 2017

[25] S.S. Dawes, L. Vidiasova, and O. Parkhimovich, "Planning and designing open government data programs: an ecosystem approach", Government Information Quarterly, vol. 33, no. 1, 2016, pp. 15-27.

[26] K. Kloeckl, O. Senn, and C. Ratti, "Enabling the Real-Time City: LIVE Singapore!", Journal of Urban Technology, vol. 19, no. 2, 2012, pp. 89-112.

[27] "Veksbarometer for Ringerike Regionen" 2018, URL: https://vekstbarometer.usn.no/assets/documents/2018.pdf [accessed 2019.12.08].

[28] "Veksbarometer for Ringerike Regionen" 2018, URL: https://vekstbarometer.usn.no/statistic/25 [accessed 2019.12.08].

[29] "Article 4 EU GDPR Definitions", URL: http://www.privacy-regulation.eu/en/article-4-definitions-GDPR.htm/ [accessed 2019.12.08].

[30] "Barometer of Telemark County in Norway", URL: http://www.telemarksbarometeret.no/ [accessed 2019.12.08]

[31] "Barometer of Hordaland and Sogn and Fjordane County in Norway", URL: http://www.naringsbarometeret.no/ [accessed 2019.12.06].

[32] A. Bruvoll and S. Pedersen, "NHOs KommuneNM 2016", Report from Vista Analyse, URL: https://legacyweb.nho.no/oppsiden/ny/html/files/VA-rapport_2016-31_NHOs_KommuneNM_2016.pdf [accessed 29.10.2018].

[33] "Innovation Norway, tool for municipalities", http://www.innovasjonnorge.no/no/regional-omstilling/verktoy/naringsvennlig-kommune/ [accessed 2019.12.08].

[34] "Confederation of Norwegian Enterprices, Municipal NM", URL:http://www.nho.no/Politikk-og-analyse/Offentlig-sektor-og-naringslivet/kommunenm/ [accessed 2019.12.08]

[35] J. Nielsen and R.L. Mack, "Usability Inspection Methods", John Wiley and Sons, New York, 1994. ISBN: 0-471-01877-5, http://dx.doi.org/10.1145/ 259963.260531.

Faster in Time and Better in Randomness Algorithms

for Matching Subjects with Multiple Controls

Hung-Jui Chang
Department of Applied Mathematics
Chung Yuan Christian University
Taoyung, Taiwan
Email: hjc@cycu.edu.tw

Yu-Hsuan Hsu
Google
Taipei, Taiwan
Email: poloo5582@gmail.com

Chih-Wen Hsueh
Department of Computer Science
and Information Engineering
National Taiwan University
Taipei, Taiwan
Email: cwhsueh@csie.ntu.edu.tw

Mei-Lien Pan, Hsiao-Mei Tsao, Da-Wei Wang, Tsan-sheng Hsu*
Institute of Information Science, Academia Sinica
Taipei, Taiwan
Email: {mlpan66, hmtsao, wdw, tshsu}@iis.sinica.edu.tw
*Corresponding Author

Abstract—In the era of learning healthcare systems and big data, observational studies play a vital role in discovering hidden (causal) associations within a dataset. To reduce bias in these observational studies, a matching step usually is adopted to randomly match each case subject with one or more control candidates. A high-quality matching algorithm, RandFlow, is proposed and compared with the commonly used – Simple Match, Matchit and Optmatch algorithms. The execution time, the memory usage, the successful matching rate, the statistical variation of relative risk, and the randomness computed employing the different algorithms are compared. The execution time of RandFlow was at least 30 times faster than commonly used methods, with at least a 66% reduction in memory usage. The variation of relative risk computed by RandFlow usually was smaller than by Simple Match. Simple Match had varying relative entropy, ranging from 0.2 to 0.95, while RandFlow almost uniformly had relative entropy close to 1. RanfFlow could find a matching so long as the maximum matching ratio was not reached. For obtaining more reliable study results, a two-phase matching is proposed. The first phase is to identify the maximum matching ratio, then is followed by matching multiple times and taking an average.

Keywords–matching; observational study; relative entropy.

I. INTRODUCTION

Observational studies are often used for investigating causal relationships [1]. Given two events α and β, researchers can analyze whether the occurrence probability of event β is affected by a previous event α. In the medical field, an event can be a diagnosis, a prescription or a treatment. To reduce bias, several approaches have been applied, one of them being matching [2]. Hence, an observational study process begins by identifying the study group G_α (those with α), matching to the control candidates $G_{\neq\alpha}$ (those without α), and then performing statistical analysis to draw a conclusion. For example, Relative Risk (RR) is used to estimate the relative risk of having β with and without the previous occurrence of α. For example, in Table I, there are $a + b$ individuals with the event α, and a

TABLE I. EXAMPLE OF STUDY GROUP AND ITS MATCHED CONTROL GROUP

	α	$\neg\alpha$
β	a	c
$\neg\beta$	b	d
Sum	$a + b$	$c + d$

of them also with the event β. The conditional probability, R_1, which denotes the probability of having β under the condition of with the event α is therefore $a/(a + b)$. Also, there are $c + d$ individuals without the event α, and c of them with the event β. The conditional probability, R_2, which denotes the probability of having β under the condition of without α is therefore $c/(c + d)$. The RR value is defined as $RR = R_1/R_2$. RR values greater than, less than, or equal to 1, respectively, indicate positive, negative or no relationships, respectively. Other statistics, such as Odds Ratio (OR), may be used instead of RR depending on the study design.

Matching is a critical step in the analysis of observational study. Generally, a matching algorithm randomly permutes the order of the input of the study case s, with the control candidate c, and then checks whether the input s-c pair can be matched, and finally matches s with K-fold eligible controls, one by one. The constant K is called the matching ratio. Some matching methods assign a propensity score to each pair [3] and return a matching with the best total score. However, if the distribution of cases is skewed, the study case may not be matchable to the required munber of controls, and so would be dropped to avoid incurring further bias. Therefore, the output matching needs to satisfy some quality criteria such as randomness and successful matching rate. In a good quality matching algorithm, a control candidate has roughly an equal chance of being matched with

any of the matchable study cases. It is desirable as well to retain as many successful matchings as possible.

There are some commonly used matching methods: for example, Simple Match [4], Matchit [5], [6], and Optmatch [7]. The first is based on a simple greedy approach using SAS and there is no proof in [4] of it being able to deliver a matching in reasonable time, and the latter two are variations of the well-known max flow algorithm [8] having a performance guarantee, but with no consideration of randomness. If the matching is only performed once along with a small matching ratio, the result may not be stable in the sense that there is the possibility that different matchings may yield fluctuating statistics such as RR or OR. To obtain a reliable result, it is better to match multiple times and take an average of all the outcomes. However, it is not practical to do repeated matching due to the heavy time consumption. Moreover in practice, the determining matching ratio is also a cloudy issue. During the past few decades, the suggested method in case-control study has been to match each subject with four or five controls [9]. It was reported that "beyond a ratio of about 4/1, little power improvement results from increasing the number of controls" [10]. However, a matching ratio of 10 or 15 has also appeared in some studies [11], [12]. In Hennessy's study, it was indicated that a higher matching ratio may be needed when the disease prevalence is low [13], which implies that the matching ratio should be data dependent [14]. To date, there has been little study investigating the issue of finding a good matching ratio.

Previous researches have focused on the impact of the matching ratio [14], and whether to use a matching or not [15]. But how to determine the matching ratio is less discussed. To resolve the above problems, we proposed a high-quality matching algorithm called *RandFlow*, which adopts the idea from maximum flow in graph theory. In RandFlow, we have added some vital functions towards raising the randomness and matching efficiency. Furthermore, we leveraged the high efficiency of RandFlow to determine the optimal matching ratio. By using RandFlow, the maximum matching ratio of each data set is calculated, and the range of the suitable matching ratio is also determined. The researcher can choose a preferred matching ratio according to the suggested range.

The remains of this paper are organized as follows: In Section II, we describe our matching algorithm, the data source used in this study, and the factor compared between different matching methods. In Section III, we show the experiment results of RandFlow and comparison between RandFlow and the original method. In Section IV, we discuss the comparison results. Finally, in Section V, we conclude this paper.

II. METHODS

The approach of our method is to formulate our problem in the well-known framework of flows in networks [8]. Hence our methods come with performance and correctness guarantees. In our study, we used Taiwan's National Health Insurance Research Database (NHIRD) [16] as the data source and examined the validity of RandFlow by three causal relations reported in the published papers [17], [18]. We then compared RandFlow with the above matching methods with regard to execution time and memory, successful matching rates, RR values and quality of randomness.

A. RandFlow Algorithm

We illustrate the idea of the original matching problem in Figure 1(a). The study cases are listed on the left-hand side, and the control candidates are listed on the right-hand side. The dashed line indicates the potential matching between a study case and a control candidate. We transform the matching problem in Figure 1(a) to the well-known max flow problem [8] in Figure 1(b) by adding one source node, one sink node, outgoing edges from the source to all study nodes, and incoming edges from the control nodes to the sink. There is a capacity constraint set on each edges where the outgoing edges of the source is K, and the rest of the edges are 1. Each edge (x, y) in E is associated a non-negative number called a weight $w(x, y)$ and, for each pair of *intermediate nodes*, namely not sources or sinks, x and y, the equality $w(x, y) = w(y, x)$ always holds. Thus w is function on E.

In a max flow problem, we assign maximum integer weights w, not exceeding the pre-assigned capacity, to the edges so that for each vertex other than the source and sink, the sum of weights on its incoming edges equals the sum of weights on its outgoing edges.

We consider a subset f of w, which we shall call a *flow*. We shall require three things for this flow to be a *legal flow*: (1) the weight of each edge is not exceeding the pre-assigned capacity, (2) for each vertex other than the source and sink, the sum of weights on its incoming edges equals the sum of weights on its outgoing edges, (3) the sum of weights on the outgoing edges of the source equals to the sum of weights on the incoming edges of the sink. A legal flow f in the max flow problem is a possible matching in the original matching problem. Those control candidates in the chosen f, whose incoming edges have nonzero weight, are the matched control cases in the original matching problem. Therefore, we can calculate the corresponding RR values of a legal flow f.

A study case S_i is matched with those control candidates C_j so that the weight of the edge from S_i to C_j is 1. The outcome is called a *max flow*. We further require that each study case has the same sum of incoming edge weights, which is called the *maximum matching ratio*, denoted by r. Thus each study case is matched with exactly r candidates, and each candidate is matched at most once. Since a max flow is to be found, r is should be as large as possible. Note that the value of r is data dependent. Each data set has its own maximum matching ratio. In addition to whether a matching of a specified size can be found efficiently or not, we also are concerned whether the resulting matching is random or not, i.e., whether each candidate has an equal chance of being selected by any case subject. Without considering constraints incurred from competitions between case subjects, we use the well-known *entropy* [19] E of the ideal distribution among all possible candidates that can be matched to a case subject. Then we measure the entropy E' of the actual distribution of candidates being found by applying the matching repeatedly say 1000 times. To quantify the quality of randomness in the matching obtained, we define the *relative entropy* to be $\frac{E'}{E}$. Some of our results infra show that finding matchings with ratio exactly r provides good randomness properties.

There are known algorithms for finding such a max flow in $O(|E||F|)$ time, where E is the set of edges and F, called *flow*, is the set of edges with weight 1 between the study cases

and candidates. The value of $|F|$ is the number of edges inside. In the general case, there are certain known algorithms, which we shall call special augmentation algorithms, for augmenting given legal flows step by step, eventually leading to legal flows taking on the maximal flow value for such a flow.

To each special augmentation algorithm corresponds an integer n. When one applies such an algorithm to a given legal flow f, which is not a maximum flow, the algorithm identifies an edge (s, c) between intermediate nodes and a set X of n edges (x, y) in the domain of f such that the nodes x are all distinct and such that each edge (a, x) is in the domain of f. Let D denote the union of three sets: the domain of f, the set $\{(s.c), (c, t)\}$ and the set of all (y, x) such that x is in X. The special augmentation algorithm acts in such a way that the restriction f' of w to D is again a legal flow. It follows from the weight property $w(x, y) = w(y, x)$ that the flow value of f' exceeds the flow value of f by the weight $w(s, c)$.

For our application here we set all weights equal to 1, and so each application of a special augmentation algorithm increases the flow value by 1.

We extend the original algorithm by finding a random augmenting flow, instead of a fixed one using a randomized version of Depth First Search (RDFS). In addition, we use a merging technique so that given two candidates C_i and C_j, they are merged if they have incoming edges from the same set of study cases. Furthermore, we randomly shuffle the ordering of study cases from the input to obtain better randomness quality. Our revised algorithm runs faster and uses less memory in practice than the original one. The technical details can be found in our technique report [20].

B. Data source

The NHIRD is a nationwide database extracted from the claim data of the National Health Insurance (NHI) program in Taiwan for research purposes. In recent years, NHIRD has been widely used to identify potential causal relationships. Our study also used NHIRD as the data source and was reviewed by the Institutional Review Board of Academia Sinica, Taiwan (approval number: AS-IRB-BM-16043). As a benchmark, we randomly generated data sets according to NHIRD data distribution. We denote P_α as the probability that an α event happens, and $P_{\beta|\neq\alpha} = P_\beta$ as the probability that a β event happens without an α event happens. And the probability that a β event happens after an α event happens is denoted as $P_{\beta|\alpha} = RR \times P_\beta$. In the following, P_α and P_β is selected from $\{0.05, 0.10, 0.15, 0.20, 0.25, 0.30, 0.35, 0.40, 0.45, 0.50\}$, and RR is selected from $\{1/2, 2/3, 4/5, 1, 5/4, 3/2, 2\}$. There are 10 possible values of P_α, 10 possible values of P_β and 7 possible values of RR. The total number of test cases is 700. For each test case, we randomly draw 1,000 person from NHIRD. According to P_α and P_β, we randomly pick α and β. We executed the RandFlow Algorithm 1,000 times and calculate the average and the standard deviation of R_1, R_2, and RR, respectively. Additionally, we selected three distinct causal relations from two published papers [17], [18]. One paper investigated the bidirectional relationship between obstructive sleep apnea (OSA) and depression [17]. The study showed the positive relationship that patients with OSA have increased risk of occurring depression, and vice versa. The other paper examined whether previous statin use in patients with stroke affects the subsequent risk of dementia [18]. The study found
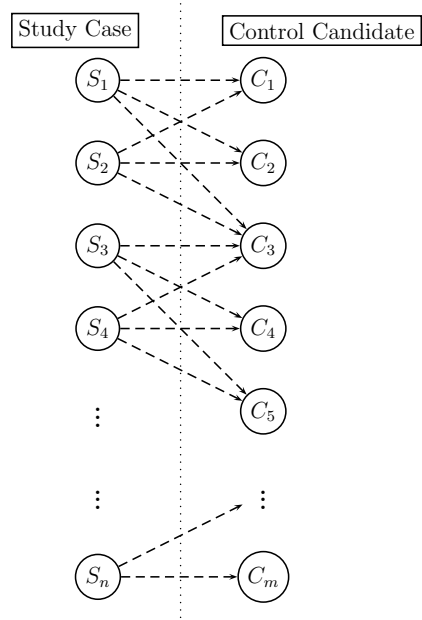
(a)
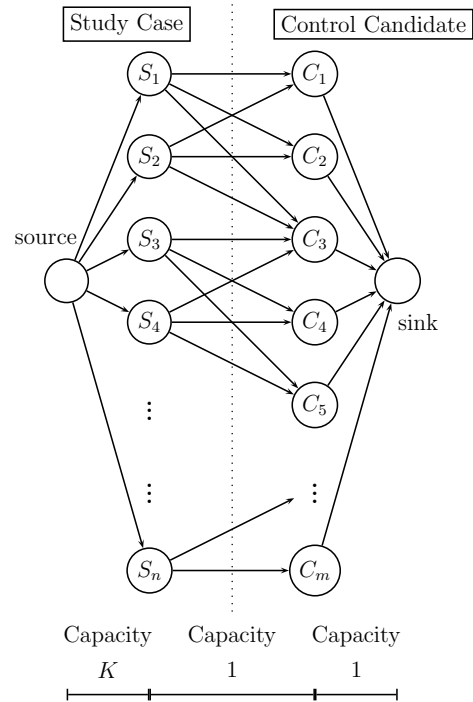
(b)

Figure 1. An example of transforming the matching problem into a flow problem.

a negative relationship in such a way that statin use in patients with stroke decreases the risk of dementia. In our study, we define an event pair to be one for which the former event affects the occurrence of the following event. Hence, the relationship between depression and subsequent OSA is denoted as Event Pair I, and the reverse is Event Pair II. The relationship between statin use in patients with stroke and subsequent dementia is Event Pair III.

C. Comparisons between matching methods

We identified the study cases of the selected event pairs according to the case criteria of [17], [18] and matched them to the control candidates by the following matching algorithms: Simple Match [4] in SAS, Matchit [5], [6] in R, Optmatch [7] in R, and RandFlow in C. Because Simple Match is the most popular matching program used in epidemiology, we implemented Simple Match using C, which we denote by Simple (C) in the sequel, in order to compare with RandFlow from a common basis. Because of software limitations and language nature, programs and packages in SAS and R run much slower and use more memory than those in C. As for Simple (C) and RandFlow, the time complexity of the former is $O(nm)$ and that of the latter is $O(n^2m)$, where n is the number of nodes and m is the number of edges in the graph. Therefore, it is expected that Simple (C) will run faster than RandFlow.

In the original studies, Event Pairs I and II were performed by the exact match method. Among these two event pairs, each study case was matched with five controls. Regarding Event Pair III, each study case was instead matched with one control by propensity score match [21]. In our study, all experiments were done by exact match. We used the ratio of control candidates to study cases on order to conjecture the maximum matching ratio. The number of total edges provided an estimate of computing time and memory consumption.

We then compared the matching methods with regard to execution time and memory usage, successful matching rates, RR values and quality of randomness. The *successful matching rate* is defined to be the percentage of matched study cases that are not dropped. We assessed the average execution time, the corresponding successful matching rates and the RR values with matching ratios from 1 to 30 (to 90 in the case of Event Pair II). To further understand the variation of RR values, we also examined the standard deviation of RR for R_1 and R_2 where R_1 and R_2, respectively, represent the risks of β occurring in the study group (G_α) and control group ($G_{\neq\alpha}$), respectively. The ratio of R_1/R_2 is RR. For quality of randomness, we calculated the relative entropy of the matched control candidates using three different matching ratios: 70%, 100% and 110% of the maximum matching ratio. The RR values and relative entropies were run 100 times, after which the average was taken. In our study, we used only Event Pair I as the benchmark to evaluate the execution time and memory usage. Because the programs implemented in C are more efficient and memory sparing, we just compared C implementations in terms of successful matching rates, RR value and quality of randomness. All the experiments were performed on a Ubuntu 14.04 system with an Intel(R) Core(TM) i7-3770 CPU 3.40 GHz, and 16 Gbytes RAM.

III. RESULTS

We present our experiment results in the follows.

TABLE II. THE STATISTIC RESULTS OF REAL RR VALUE AND THE ESTIMATED RR VALUE OF RANDFLOW.

Real RR	Estimated RR	Δ	Variance	STD
0.50 (1/2)	0.462	0.038	0.021	0.144
0.66 (2/3)	0.658	0.002	0.033	0.182
0.80 (4/5)	0.854	0.054	0.041	0.202
1.00 (1/1)	1.016	0.016	0.029	0.169
1.25 (5/4)	1.259	0.009	0.049	0.209
1.50 (3/2)	1.542	0.042	0.056	0.236
2.00 (2/1)	2.049	0.049	0.105	0.325

A. General result of the randomly sampled data

Figure 2 shows the result of the randomly generated data. The x-axis denotes the real RR value, and the y-axis denotes the estimated RR value, which is calculated by RandFlow. Each point in Figure 2 represents one data set. For each real RR value, there are 100 test data sets. The results show RandFlow can get an estimated RR value very close to the real RR value. The statistic results are summarized in Table II. The first and second column denotes the real RR value and the estimated RR value. The third to fifth column denotes the absolute error between the real and the estimated RR value, the variance of the estimated RR value and the standard deviation of the estimated RR value. The experiment results show the absolute error RandFlow Algorithm is less than 0.06 and the variance and standard deviation is only 0.10 and 0.33, respectively.

B. General information of the selected event pairs

Table III shows the general information of the selected event pairs from the original papers together with our results, including the number of controls/control candidates, the ratio of control candidates to study cases, and the maximum matching ratio.

Among these event pairs, the greatest number of study cases was found in Event Pair I. With such a larde number of study cases, there were a total of more than 149 million edges generated while matching using RandFlow. We speculated that the maximum matching ratio would be different among the event pairs since it turned out to be the ratios 11, 51 and zero, respectively, for Event Pair I, II and III, respectively. In addition, these event pairs covered both positive and negative relationships. As a result, we believed that they can serve as representatives for testing matching quality - Event Pair I especially for execution time and memory usage comparison.

C. Execution time and memory usage

The benchmark experiments used the Event-Pair I dataset to measure the performance. Table IV shows the comparison of average execution times and memory usage among the selected matching methods. We measure the amount of memory usage at a matching ratio equals 1. And we only shows two digits after decimal point for execution time. Because of the different time complexity, Simple (C) ran faster than RandFlow. Regarding memory usage, these two methods are comparable. However, the average execution times of Matchit increased roughly linearly with the increase of the required matching ratio (Figure 3).
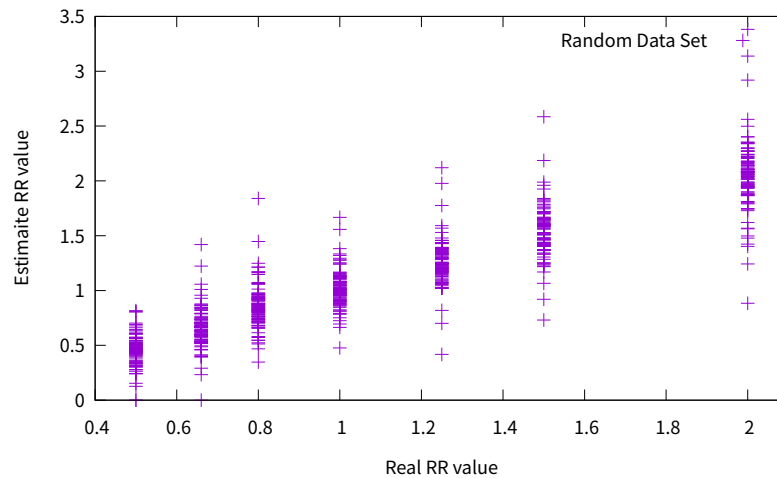
Figure 2. The distribution of real RR value and the estimated RR value of RandFlow.

TABLE III. GENERAL INFORMATION OF THE SELECTED EVENT PAIRS.

	Event Pair I	Event Pair II	Event Pair III
Original results			
No. study cases	27,073	6,427	5,527
No. control cases	135,365	32,135	5,527
Matching ratio	5	5	1
Our results			
No. control candidates	562,707	619,904	9,102
Control candidates/Study cases	≈21	≈97	≈2
Maximum matching ratio	11	51	0
Total edge	149,676,628	38,629,676	404,835

TABLE IV. AVERAGE EXECUTION TIMES (IN SECONDS[B]) AND MEMORY USAGE (IN GB) OF EACH METHOD.

Method	Lang.	Average execution times at different matching ratio(sec)							Memory
		1	5	10	15	20	25	30	usage (Gb)
Simple Match	SAS	1221.00	959.00	894.00	859.00	2350.00	1228.00	1078.00	1.039
Optmatch	R	8473.28	9132.34	9676.67	9407.05	9605.70	9037.06	8399.11	0.277
MatchIt	R	182.81	928.57	1956.58	2890.76	3891.28	4821.20	5924.35	0.232
Simple (C)	C	0.51	0.51	0.51	0.51	0.51	0.51	0.51	0.068
RandFlow	C	7.06	7.03	7.04	5.97	5.56	5.19	4.71	0.070

D. RR values and Successful matching rates

Flow-based matching methods in nature continue matching until they use up all the matchable control candidates. They are expected to have the same traits in terms of RR value variation and successful matching rate. Hence, in Section III, we only show the comparisons between Simple (C) and RandFlow.

Figure 4 shows the comparison results between RandFlow and Simple Match. Each column denotes the experiment results of different event pair, and each row denotes one specific comparison. The first row (Figure 4(a)-4(c)) shows the comparison of the RR value under different matching ratio. The second row (Figure 4(d)-4(f)) shows the comparison of the standard deviation of the RR value under different matching ratio. And the last row (Figure 4(g)-4(i)) shows the standard deviation of R1 and R2, which are performed by RandFlow under different matching ratio.

Overall, the average RR values of Simple (C) were greater than the values of RandFlow. In both methods, the average RR values were fairly stable while the matching ratio was small, and then gradually decreased when the matching ratio exceeded a certain value. In RandFlow, the decline occurred at the maximum matching ratio. By contrast, the decline of Simple (C) occurred earlier than that (Figure 4(a) and 4(b)). In the case of a negative relationship in Event Pair III, the average RR values increased rather than decreased (Figure 4(c)).

Generally speaking, the variation of RR values of Simple (C) was more unstable than that of RandFlow. In both methods, the variation of RR values steadily decreased and then turned up after a certain matching ratio. The least variation of the RR values of RandFlow occurred right at the maximum matching ratio. That of Simple (C) occurred before the maximum matching ratio (Figure 4(d)-4(f)).
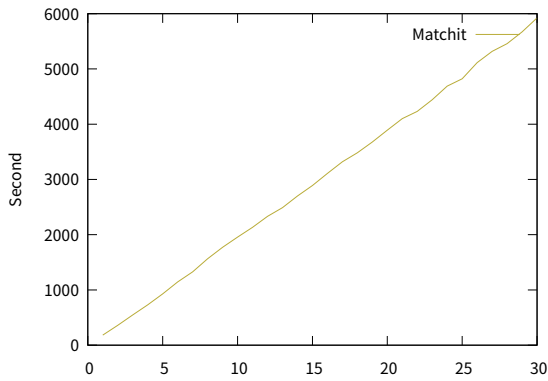
(a) Matchit
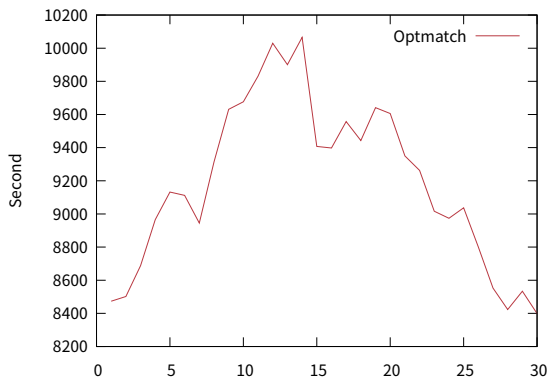
(b) Optmatch
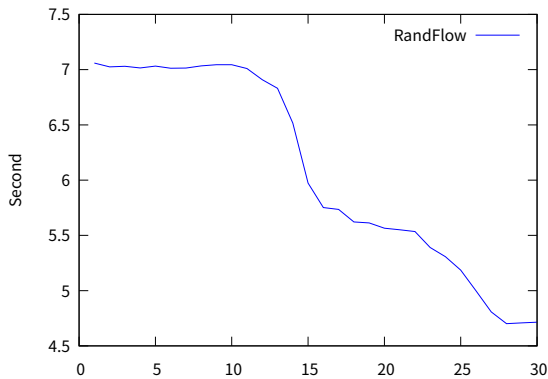
(c) RandFlow

Figure 3. Average execution times of the flow-based matching methods.

Since RR is calculated as R_1 divided by R_2, we examined the variation of R_1 and R_2 in RandFlow to survey further where the RR variation originates. When the matching ratio was less than the maximum matching ratio, no study cases were dropped; thus, the standard deviation of R_1 remained zero. On the other hand, the standard deviation of R_2 decreased with matching ratio, until it reached its maximum. As the size of the control group increased up to a certain number, the standard deviation of R_2 continued relatively small and steady. Beyond the maximum, the standard deviation of R_1 surged, resulting from the dropping of study cases (Figure 4(g)-4(i)).

Figure 5 shows the comparison of successful matching rates between Simple (C) and RandFlow. Because Simple (C) is based on a simple greed algorithm, its matching results may vary. We used both the minimal (Simple_min) and the maximal (Simple_max) results from the 100 trials for comparison. Whether or not we used the minimal or the maximal result from Simple (C), the successful matching rates dropped before reaching the maximum matching ratio, whereas those of RandFlow remained at 100%. At any fixed matching ratio, RandFlow had the highest successful matching rates. Although Simple (C) ran faster than RandFlow, when the execution time was fixed it failed to achieve the successful matching rate of RandFlow.

E. Quality of randomness

Optmatch and Matchit are both flow-based matching methods and do not randomly shuffle the input graph. In other words, their matched results remain unchangeable with no randomness at all. By contrast, we implemented RandFlow with inputting random graph and RDFS to enhance the quality of randomness. In this section, we present a comparison of quality of randomness between RandFlow and Simple (C).

Figure 6 shows the relative entropy of the chosen control candidates of the study cases in Event Pair I and Event pair II. The first column shows the result of Event Pair I, and the second column shows the result of Event Pair II. Each row denotes different matching ratio, the first row, the second row, and the last row shows the experiment result with matching ratio equals to 70%, 100%, and 110% of the maximum matching ratio, respectively.

Figure 6 shows that Randflow had better quality of randomness than Simple (C). The estimated relative entropy of RandFlow was about 1 and generally higher than that of Simple (C). In addition, RandFlow had consistently stable entropy for all matching ratio and study cases. Even when the ratio was set at 110% of the maximum matching ratio, the relative entropy of Randflow only decreased slightly. For those study cases having small sets of control candidates, the relative entropy of Randflow remained high. By contrast, the relative entropy of Simple (C) fluctuated widely as the matching ratio increased. For those study cases having less matchable control candidates, that of Simple (C) plunged.

IV. DISCUSSION

In this study, we adopted maximum flow theory to develop a highly efficient and good-quality matching method, *RandFlow*, for matching subjects with multiple controls. This method can accomplish difficult matching tasks, such as matching 20 thousand study cases to each to 30 controls within a few seconds. Compared with the most popular matching method,
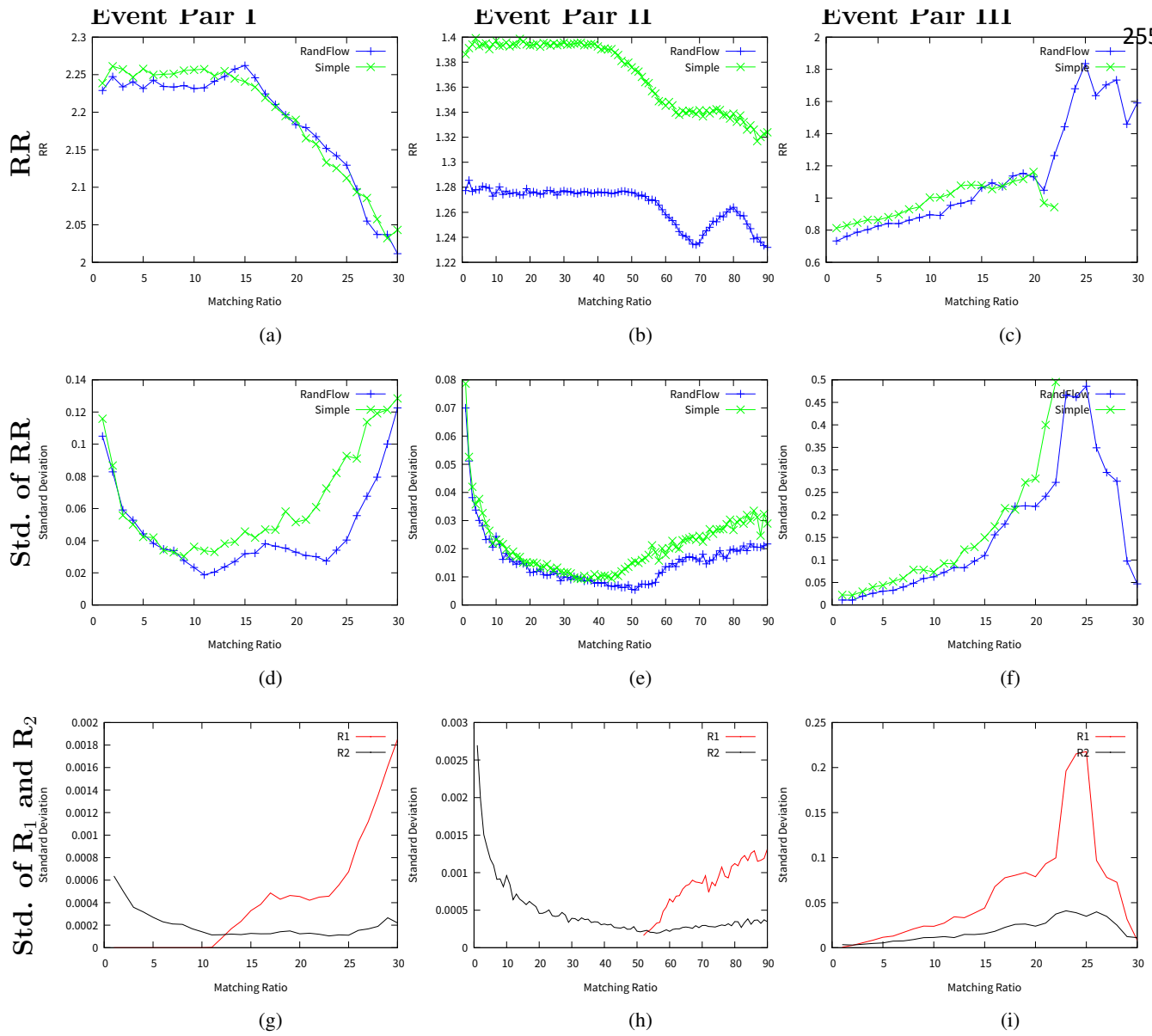
Figure 4. RR values and standard deviation of RR, R_1 and R_2 of Simple (C) and RandFlow.

Simple Match (on a SAS platform), it consumed merely 0.5-2.4% of execution time and 7% of memory usage. Among the flow-based matching methods, Optmatch and RandFlow are much alike in terms of execution time versus matching ratio. In both of these methods, the average execution times remained about the same until the required matching ratio exceeded the maximum matching ratio, after which they decreased because more and more case subjects were subsequently dropped.

Most importantly, RandFlow exhibited a good quality of randomness and rather than dropping study cases found a matching whenever such a matching existed. Matching is used to cause study cases and controls to have similar distributions across confounding variables. During the matching process, the controls are expected to be randomly selected from all the control candidates. Anything that may affect the sampling design, such as the dropping of cases, should be avoided. Our study used relative entropy to quantify randomness, and then

verified that RandFlow had a good quality of randomness. The randomness of RandFlow does not vary with the chosen matching ratios since it is no more than the maximum ratio. With regards to successful matching rate, RandFlow outperformed simple greedy algorithms due to the nature of those algorithms. Overall, RandFlow surpassed those commonly used matching methods. It is not only a highly efficient matching method but also includes processes for avoiding undesirable biases during matching.

The matching ratio is data dependent and should be set differentially at the maximum matching ratio to obtain consistent results. During the past few decades, the suggested matching method of case-control study was to match each case subject with four or five controls. Previous studies had indicated that a higher matching ratio might be desired [10], [13], [14]. Beyond the previous studies, we tested three distinct data sets and performed matching multiple times at a range

(a) Event Pair I

(b) Event Pair II

(c) Event Pair III

Figure 5. Successful matching rate of Simple (C) and RandFlow. Simple_min and Simple_max represent the minimal and the maximal matching rate from the 100 trials run by Simple (C).

h requested is no more than the maximum matching ratio w, then we have many possible different matchings. From the law of large numbers, the RR value calculated from many instances must be stable and close to the real average case. If h is more than w, then we do not have many choices in selecting the pairings. The deviation of the computed RR tends to be higher than in the former case. Therefore, rather than using an empirical fixed matching ratio for any given study, we suggest matching at the maximum matching ratio multiple times and then taking an average for consistent results.

RandFlow being an exact matching has an inherent limitation: that of being unable to match some study cases with the required number of controls when the distribution of the confounding variables is skewed. In the extreme case, even a 1:1 match cannot be reached; thus, the RR values will be unstable at any matching ratio. In this circumstance, in order to obtains reliable results, other matching methods should be considered.

V. Conclusions

In this study, we developed a highly efficient matching method and demonstrated that it provides a good quality of randomness. From our experiments, we further concluded that the matching ratio is data dependent and should be set differentially at the maximum matching ratio. For future study, we suggest that matching should be done in two phases. The first phase is to identify the maximum matching ratio. Then, the second phase is to carry out matching using the maximum matching ratio several times and taking an average statistics. Using this two-phase matching, researchers can obtain stable results and accordingly draw unbiased study conclusions.

References

[1] H.-J. Chang, Y.-H. Hsu, C.-W. Hsueh, and T.-s. Hsu, "Efficient qualitative method for matching subjects with multiple controls," in Proceedings of the Fifth International Conference on Big Data, Small Data, Linked Data and Open Data, ALLDATA 2019.

[2] E. A. Stuart, "Matching methods for causal inference: A review and a look forward," Statistical science: a review journal of the Institute of Mathematical Statistics, vol. 25, no. 1, 2010, p. 1.

[3] P. R. Rosenbaum and D. B. Rubin, "Constructing a control group using multivariate matched sampling methods that incorporate the propensity score," The American Statistician, vol. 39, no. 1, 1985, pp. 33–38.

[4] H. Kawabata, M. Tran, and P. Hines, "Using SAS® to match cases for case control studies," in Proceeding of the Twenty-Ninth Annual SAS® Users Group International Conference, vol. 29, 2004, pp. 173–29.

[5] D. E. Ho, K. Imai, G. King, and E. A. Stuart, "Matching as nonparametric preprocessing for reducing model dependence in parametric causal inference," Political analysis, vol. 15, no. 3, 2007, pp. 199–236.

of matching ratios. In our experiments, we found that the maximum matching ratio varied with the input data set and the least variation of RR values always occurred when we set the matching ratio to be the maximum. This can be explained from the perspective of graph theory. If the matching ratio
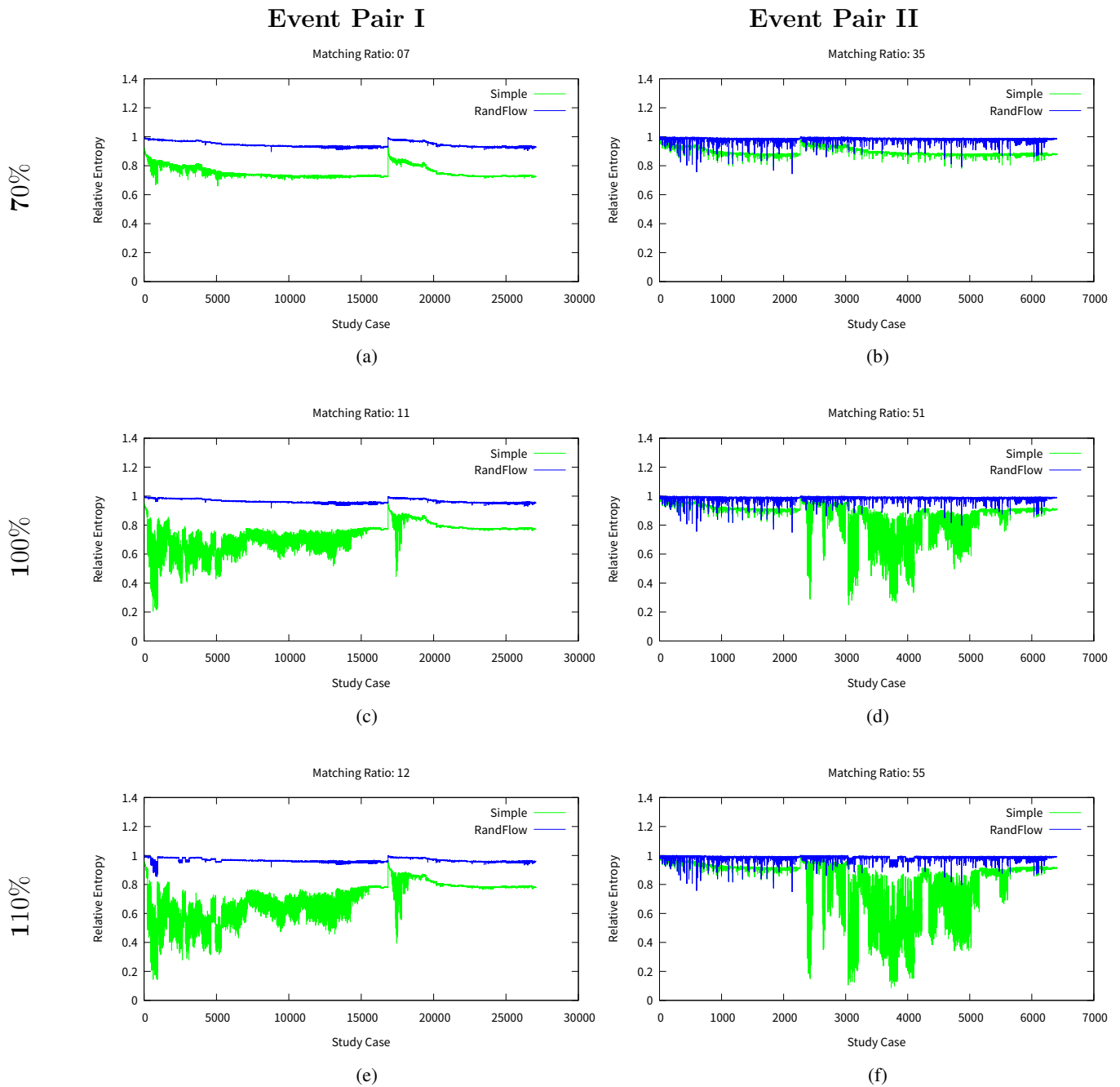
Figure 6. Relative entropy of Simple (C) and RandFlow in Event Pair I, II. Relative entropy of Event Pair I and II was tested at 70%, 100% and 110% of the maximum matching ratio in 100 trials.

[6] ——, "Matchit: Nonparametric preprocessing for parametric causal inference," Journal of Statistical Software, 2007.

[7] B. B. Hansen, "Full matching in an observational study of coaching for the SAT," Journal of the American Statistical Association, vol. 99, no. 467, 2004, pp. 609–618.

[8] L. R. F. Jr and D. R. Fulkerson, "Maximal flow through a network," Canadian journal of Mathematics, vol. 8, no. 3, 1956, pp. 399–404.

[9] S. Wacholder, D. T. Silverman, J. K. McLaughlin, and J. S. Mandel, "Selection of controls in case-control studies: Iii. design options," American journal of epidemiology, vol. 135, no. 9, 1992, pp. 1042–1050.

[10] D. A. Grimes and K. F. Schulz, "Compared to what? finding controls for case-control studies," The Lancet, vol. 365, no. 9468, 2005, pp. 1429–1433.

[11] M.-L. Pan, L.-R. Chen, H.-M. Tsao, and K.-H. Chen, "Relationship between polycystic ovarian syndrome and subsequent gestational diabetes mellitus: a nationwide population-based study," PloS one, vol. 10, no. 10, 2015, p. e0140544.

[12] K.-J. Tien, C.-W. Chou, S.-Y. Lee, N.-C. Yeh, C.-Y. Yang, F.-C. Yen, J.-J. Wang, and S.-F. Weng, "Obstructive sleep apnea and the risk of atopic dermatitis: A population-based case control study," PloS one, vol. 9, no. 2, 2014, p. e89656.

[13] S. Hennessy, W. B. Bilker, J. A. Berlin, and B. L. Strom, "Factors

influencing the optimal control-to-case ratio in matched case-control studies," American Journal of Epidemiology, vol. 149, no. 2, 1999, pp. 195–197.

[14] K. J. Rothman, S. Greenland, and T. L. Lash, Modern epidemiology. Lippincott Williams & Wilkins, 2008.

[15] T. Faresjö and Å. Faresjö, "To match or not to match in epidemiological studiesxsame outcome but less power," International journal of environmental research and public health, vol. 7, no. 1, 2010, pp. 325–332.

[16] M. o. H. National Health Insurance Administration and R. Welfare, Taiwan, "National health insurance research database, Taiwan." retrieved: December, 2019. [Online]. Available: http://nhird.nhri.org.tw/en/index.htm

[17] M.-L. Pan, H.-M. Tsao, C.-C. Hsu, K.-M. Wu, T.-s. Hsu, Y.-T. Wu, and G.-C. Hu, "Bidirectional association between obstructive sleep apnea and depression: A population-based longitudinal study," Medicine, vol. 95, no. 37, 2016, p. e4833.

[18] M.-L. Pan, C.-C. Hsu, Y.-M. Chen, H.-K. Yu, and G.-C. Hu, "Statin use and the risk of dementia in patients with stroke: A nationwide population-based cohort study," Journal of Stroke and Cerebrovascular Diseases, 2018.

[19] C. E. Shannon, "A mathematical theory of communication," ACM SIGMOBILE Mobile Computing and Communications Review, vol. 5, no. 1, 2001, pp. 3–55.

[20] H.-J. Chang, Y.-H. Hsu, C.-W. Hsueh, and T.-s. Hsu, "Efficient randomized algorithms for large-scaled exact matching with multiple controls: Implementation and applications," Institution of Information Science, Academia Sinica, Taiwan, Tech. Rep. TR-IIS-17-005, 2017.

[21] L. S. Parsons, "Reducing bias in a propensity score matched-pair sample using greedy matching techniques," The Twenty-Sixth Annual SAS Users Group International Conference, vol. 21426, 01 2001.

Data Loss in RAID-5 and RAID-6 Storage Systems with Latent Errors

Ilias Iliadis

IBM Research – Zurich

8803 Rüschlikon, Switzerland

Email: ili@zurich.ibm.com

Abstract—**Storage systems employ redundancy and recovering schemes to protect against device failures and latent sector errors, as well as to enhance reliability. The effectiveness of these schemes has been evaluated based on the Mean Time to Data Loss** (MTTDL) **and the Expected Annual Fraction of Data Loss** (EAFDL) **metrics. The reliability degradation due to device failures has been assessed in terms of both these metrics, but the adverse effect of latent errors has been assessed only in terms of the** MTTDL **metric. This article addresses the issue of evaluating the amount of data losses caused by latent errors. It presents a methodology for obtaining** MTTDL **and** EAFDL **of RAID-5 and RAID-6 systems analytically in the presence of unrecoverable or latent errors. A theoretical model capturing the effect of independent latent errors and device failures is developed, and closed-form expressions are derived for the metrics of interest.**

Keywords–Storage; Unrecoverable or latent sector errors; Reliability analysis; MTTDL; EAFDL; RAID; MDS codes; stochastic modeling.

I. INTRODUCTION

Today's large-scale data storage systems use data redundancy schemes to recover data lost due to device and component failures as well as to enhance reliability [1]. Erasure coding schemes are deployed that provide high data reliability as well as high storage efficiency. Special cases of erasure codes are the replication schemes and the Redundant Arrays of Inexpensive Disks (RAID) schemes, such as RAID-5 and RAID-6, which have been deployed extensively in the past thirty years [2-5]. The effectiveness of these schemes has been evaluated based on the Mean Time to Data Loss (MTTDL) [2-11] and, more recently, the Expected Annual Fraction of Data Loss (EAFDL) reliability metrics [12][13][14]. The latter metric was introduced because Amazon S3 considers the durability of data over a given year [15], and, similarly, Facebook [16], LinkedIn [17] and Yahoo! [18] consider the amount of lost data in given periods.

The reliability of storage systems is also degraded by the occurrence of unrecoverable sector errors, that is, errors that cannot be corrected by the standard sector-associated error-correcting code (ECC) nor by the re-read mechanism of hard-disk drives (HDDs). These sector errors are latent because their existence is only discovered when there is an attempt to access them. Once an unrecoverable or latent sector error is detected, it can usually be corrected by the RAID capability. However, if this is not feasible, these sectors are permanently lost, leading to an unrecoverable failure. Consequently, unrecoverable errors do not necessarily lead to unrecoverable failures. The effect of latent errors is quite pronounced in higher-capacity HDDs and storage nodes because of the high frequency of these errors [19-23]. The risk of permanent loss of data rises in the presence of latent errors.

Analytical reliability expressions for MTTDL that take into account the effect of latent errors have been obtained predominately using Markovian models, which assume that component failure and rebuild times are independent and exponentially distributed [8][21][22][24]. The effect of latent errors on MTTDL of erasure-coded storage systems for the practical case of non-exponential failure and rebuild time distributions was assessed in [23].

In this article, we consider the effect of latent errors not only on MTTDL, but also on the amount of lost data for the case of non-exponential failure and rebuild time distributions. Clearly, when a data loss occurs, the amount of data lost due to a device failure is much larger than the amount of sectors lost due to latent errors. We present a non-Markovian methodology for deriving the MTTDL and EAFDL metrics analytically for the case of RAID-5 and RAID-6 systems. We extend the methodology developed in prior work [12][13] to assess MTTDL and EAFDL of storage systems in the absence of latent errors. The validity of this methodology for accurately assessing the reliability of storage systems was confirmed by simulations in several contexts [4][9][12][25]. It has been demonstrated that theoretical predictions of the reliability of systems comprising highly reliable storage devices are in good agreement with simulation results. Consequently, the emphasis of the present work is on theoretically assessing the effect of latent errors on system reliability. This is the first work to study the effect of latent errors on EAFDL.

The key contributions of this article are the following. We consider the reliability of RAID storage systems that was assessed in our earlier work [1] for RAID-5 systems. We now extend our previous work by considering RAID-6 systems, which tolerate two device failures. We derive analytically the MTTDL and EAFDL reliability metrics. We subsequently establish that, for typical frequencies of sector errors, the probability of encountering an unrecoverable failure is much greater than that of encountering a device failure, which degrades MTTDL, but the EAFDL is practically unaffected in this range.

The remainder of the article is organized as follows. Section II describes the storage system model and the corresponding parameters considered. Section III considers the unrecoverable or latent errors and the frequency of their occurrence. Section IV presents the general framework and methodology

for deriving the MTTDL and EAFDL metrics analytically for the case of RAID systems and in the presence of latent errors. Closed-form expressions for relevant reliability metrics, such as the probability of data loss and the amount of data loss, are derived in Sections V and VI for RAID-5 and RAID-6 systems, respectively. Section VII presents numerical results demonstrating the effectiveness of the RAID-5 and RAID-6 schemes for improving system reliability and the adverse effect of unrecoverable or latent errors on the probability of data loss and on the MTTDL and EAFDL reliability metrics. Section VIII provides a discussion concerning the results obtained. Finally, we conclude in Section IX.

II. STORAGE SYSTEM MODEL

The storage system considered here comprises n storage devices (nodes or disks), where each device stores an amount c of data such that the total storage capacity of the system is $n\,c$. User data is divided into blocks (or symbols) of a fixed size s (e.g., sector size of 512 bytes) and complemented with parity symbols to form codewords.

A. Redundancy

We consider an $(m, l) = (N, N - 1)$ maximum distance separable (MDS) erasure code, which is a mapping from $N-1$ user-data symbols to a set of N symbols, called a codeword, having the property that any subset containing $N-1$ of the N symbols of the codeword can be used to decode (reconstruct, recover) the codeword. A single parity symbol is computed by using the XOR operation on $l = N - 1$ user-data symbols to form a codeword with $m = N$ symbols in total. Such a scheme can tolerate a single erasure anywhere in the codeword. The N symbols of each codeword are stored on N distinct devices. More specifically, this scheme is used by the popular RAID-5 system, in which the n devices are arranged in groups (or arrays) of N devices, one of which is redundant [2][3]. The storage system therefore comprises n/N RAID-5 arrays, where each array has the ability to tolerate one device failure.

We also consider an $(m, l) = (N, N - 2)$ MDS erasure code, which is a mapping from $N - 2$ user-data symbols to a set of N codeword symbols having the property that any subset containing $N - 2$ of the N symbols of the codeword can be used to decode (reconstruct, recover) the codeword. A codeword contains $l = N - 2$ user-data symbols and two parity symbols for a total of $m = N$ symbols. Such a scheme can tolerate two erasures anywhere in the codeword. The N symbols of each codeword are stored on N distinct devices. More specifically, this scheme is used by the popular RAID-6 system, in which the n devices are arranged in groups (or arrays) of N devices, two of which are redundant [3]. The storage system therefore comprises n/N RAID-6 arrays, where each array has the ability to tolerate two device failures.

The storage efficiency $se^{(\text{RAID})}$ of the system is [11][13]

$$se^{(\text{RAID})} = \frac{l}{m} = \begin{cases} \frac{N-1}{N}, & \text{for RAID 5} \\ \frac{N-2}{N}, & \text{for RAID 6} \end{cases} \quad (1)$$

and the amount of user data U stored in the system is [13]

$$U = se^{(\text{RAID})}\, n\, c = \frac{l\, n\, c}{m}. \quad (2)$$

TABLE I.   NOTATION OF SYSTEM PARAMETERS

Parameter	Definition
n	number of storage devices
c	amount of data stored on each device
l	number of user-data symbols per codeword ($l \geq 1$)
m	total number of symbols per codeword ($m > l$)
(m, l)	MDS-code structure
s	symbol size
N	number of devices in a RAID array ($N = m$)
b	average reserved rebuild bandwidth per device
R	time required to read (or write) an amount c of data at an average rate b from (or to) a device
$F_R(.)$	cumulative distribution function of R
$F_\lambda(.)$	cumulative distribution function of device lifetimes
P_{bit}	Probability of an unrecoverable bit error
$se^{(\text{RAID})}$	storage efficiency of RAID redundancy scheme ($se^{(\text{RAID})} = l/m$)
U	amount of user data stored in the system ($U = se^{(\text{RAID})}\, n\, c$)
C	number of codewords stored in a RAID array ($C = c/s$)
μ^{-1}	mean time to read (or write) an amount c of data at an average rate b from (or to) a device ($\mu^{-1} = E(R) = c/b$)
λ^{-1}	mean time to failure of a storage device ($\lambda^{-1} = \int_0^\infty [1 - F_\lambda(t)]dt$)
P_s	Probability of an unrecoverable sector (symbol) error
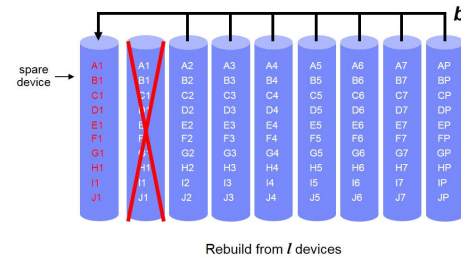
Figure 1.   Rebuild for a RAID-5 array with $N = m = 8$ and $l = 7$.

Also, the number C of symbols in a device or, equivalently, the number of codewords in a RAID array is

$$C = \frac{c}{s}. \quad (3)$$

Our notation is summarized in Table I. The parameters are divided according to whether they are independent or derived, and are listed in the upper and lower part of the table, respectively.

B. Codeword Reconstruction

When a storage device of an array fails, the C codewords stored in the array lose one of their symbols. Subsequently, the system starts to reconstruct the lost codeword symbols using the surviving symbols of the affected codewords. We assume that device failures are detected instantaneously, which immediately triggers the rebuild process. A certain proportion of the device bandwidth is reserved for data recovery during the rebuild process, where b denotes the actual average reserved rebuild bandwidth per device. This bandwidth is usually only a fraction of the total bandwidth available at each device, the remaining bandwidth being used to serve user requests.

The rebuild process attempts to restore the codewords of the affected array sequentially. The lost symbols are reconstructed directly in a spare device as shown in Figure 1. Decoding and re-encoding of data are assumed to be done on the fly, so the reconstruction time is equal to the time taken to read and write the required data to the spare device. Consequently, the time required to recover the amount c of

lost data is equal to the time R required to read (or write) an amount c of data from (or to) a device. In particular, $1/\mu$ denotes the average time required to read (or write) an amount c of data from (or to) a device, which is defined by

$$\frac{1}{\mu} \triangleq E(R) = \frac{c}{b} \, . \tag{4}$$

C. Failure and Rebuild Time Distributions

We adopt the model and notation considered in [13]. The lifetimes of the n devices are assumed to be independent and identically distributed, with a cumulative distribution function $F_\lambda(.)$ and a mean of $1/\lambda$. We consider real-world distributions, such as Weibull and gamma, as well as exponential distributions that belong to the large class defined in [25]. The storage devices are characterized as *highly reliable* in that the ratio of the mean time $1/\mu$ to read all contents of a device (which typically is on the order of tens of hours), to the mean time to failure of a device $1/\lambda$ (which is typically on the order of thousands of hours) is very small, that is,

$$\frac{\lambda}{\mu} = \frac{\lambda c}{b} \ll 1 \, . \tag{5}$$

We consider storage devices whose cumulative distribution function F_λ satisfies the condition

$$\mu \int_0^\infty F_\lambda(t)[1 - F_R(t)]dt \ll 1, \quad \text{with } \frac{\lambda}{\mu} \ll 1 \, , \tag{6}$$

where $F_R(.)$ is the cumulative distribution function of the rebuild time R. Then the MTTDL and EAFDL reliability metrics tend to be insensitive to the device failure distribution, that is, they depend only on its mean $1/\lambda$, but not on its density $F_\lambda(.)$ [13].

III. Data Loss due to Unrecoverable Errors

The reliability of RAID systems is affected by the occurrence of unrecoverable or latent errors. Let P_{bit} denote the unrecoverable bit-error probability. According to the specifications, P_{bit} is equal to 1×10^{-15} for SCSI drives and 1×10^{-14} for SATA drives [8]. Assuming that bit errors occur independently over successive bits, the unrecoverable sector (symbol) error probability P_s is

$$P_s = 1 - (1 - P_{\text{bit}})^s \, , \tag{7}$$

with s expressed in bits. Assuming a sector size of 512 bytes, the equivalent unrecoverable sector error probability is $P_s \approx P_{\text{bit}} \times 4096$, which is 4.096×10^{-12} in the case of SCSI and 4.096×10^{-11} in the case of SATA drives. However, empirical field results suggest that the actual values can be orders of magnitude higher, reaching $P_s \approx 5 \times 10^{-9}$ [26].

IV. Derivation of MTTDL and EAFDL

The MTTDL metric assesses the expected time until some data can no longer be recovered and therefore is lost forever, whereas the EAFDL assesses the fraction of stored data that is expected to be lost by the system annually. We briefly review the general methodology for deriving the MTTDL and EAFDL metrics presented in [12]. This methodology does not involve Markovian analysis and holds for general failure time distributions, which can be exponential or non-exponential,

such as the Weibull and gamma distributions that satisfy condition (6).

At any point in time, the system can be thought to be in one of two modes: normal or rebuild mode. During normal mode, all devices are operational and all data in the system has the original amount of redundancy. Any symbols encountered with unrecoverable or latent errors are usually corrected by the RAID capability. However, it may no be possible to recover multiple unrecoverable errors in a codeword, which therefore leads to data loss. A transition from normal to rebuild mode occurs when a device fails; we refer to the device failure that causes this transition as a *first device* failure. During rebuild mode, an active rebuild process attempts to restore the lost data in a spare device, which eventually leads the system either to a data loss (DL) with probability P_{DL} or back to the original normal mode by restoring initial redundancy, which occurs with probability $1 - P_{\text{DL}}$.

Let T be a typical interval of a fully operational period, that is, the interval from the time t that the system is brought to its original state until a subsequent first device failure occurs. For a system comprising n devices with a mean time to failure of a device $1/\lambda$, the expected duration of T is [12]

$$E(T) = 1/(n\,\lambda) \, , \tag{8}$$

and MTTDL is

$$\text{MTTDL} \approx \frac{E(T)}{P_{\text{DL}}} = \frac{1}{n\,\lambda\,P_{\text{DL}}} \, . \tag{9}$$

The EAFDL is obtained as the ratio of the expected amount $E(Q)$ of lost user data, normalized to the amount U of user data, to the expected duration of T [12, Equation (9)]:

$$\text{EAFDL} \approx \frac{E(Q)}{E(T) \cdot U} \overset{(8)}{=} \frac{n\,\lambda\,E(Q)}{U} \overset{(2)}{=} \frac{m\,\lambda\,E(Q)}{l\,c} \, , \tag{10}$$

with $E(T)$ and $1/\lambda$ expressed in years.

The expected amount $E(H)$ of lost user data, given that data loss has occurred, is determined by [12, Equation (8)]:

$$E(H) = \frac{E(Q)}{P_{\text{DL}}} \, . \tag{11}$$

It follows from (9) and (10) that the derivation of the MTTDL and EAFDL metrics requires the evaluation of P_{DL} and $E(Q)$, respectively. These quantities are derived using the direct path approximation [4][25][27], which, under conditions (5) and (6), accurately assesses the reliability metrics of interest [11][12][25][28].

V. RAID-5 Systems

Here we derive the reliability metrics for a RAID-5 system. When a storage device of a RAID-5 array fails, the C codewords stored in the array lose one of their symbols. Using the direct-path-approximation methodology, we proceed by considering only the subsequent potential data losses and device failures related to the affected array.

TABLE II.   NOTATION OF VARIABLES

Parameter	Definition
I	number of codeword symbols with unrecoverable errors
L	number of codeword symbols lost
q	probability that a codeword can be restored
P_{DL}	probability of data loss
P_{UF}	probability of unrecoverable failures
S	number of lost symbols
Q	amount of lost user data
H	amount of lost user data, given that data loss has occurred

A. One Device Failure

The rebuild process attempts to restore the C codewords of the affected array sequentially. Let us consider such a codeword and let L_1 be the number of symbols permanently lost and I_1 be the number of symbols in the codeword with unrecoverable errors, as listed in Table II, where the subscript denotes the number of device failures. Owing to the independence of symbol errors, I_1 follows a binomial distribution with parameter P_s, the probability that a symbol has a latent (unrecoverable) error. Thus,

$$P(I_1 = i) = \binom{m-1}{i} P_s^i (1-P_s)^{m-1-i}, \text{ for } i = 0, \ldots, m-1 \,, \tag{12}$$

such that

$$E(I_1) = \sum_{i=1}^{m-1} i\, P(I_1 = i) \,=\, (m-1)\, P_s \,. \tag{13}$$

Clearly, the symbol lost due to the device failure can be corrected by the RAID-5 capability only if the remaining $m-1$ symbols can be read. Thus, $L_1 = 0$ if and only if $I_1 = 0$. Using (12), the probability q_1 that a codeword can be restored is

$$q_1 = P(L_1 = 0) = P(I_1 = 0) = (1-P_s)^{m-1} \,, \tag{14}$$

which for very small values of P_s implies that

$$q_1 \approx \begin{cases} 1 - (m-1)\, P_s \,, & \text{for } P_s \ll \frac{1}{m-1} \\ 0 \,, & \text{for } P_s \gg \frac{1}{m-1} \,. \end{cases} \tag{15}$$

Note that if a codeword cannot be restored, then at least one of its l user-data symbols is lost. We now deduce that the conditional probability $P_{\mathrm{UF}|1}$ of encountering an unrecoverable failure during the rebuild process of the C codewords given one device failure is

$$P_{\mathrm{UF}|1} \,=\, 1 - q_1^C \tag{16}$$

$$\overset{(14)}{=} 1 - (1-P_s)^{(m-1)\,C} \,. \tag{17}$$

Furthermore, such an unrecoverable failure entails the loss of user data. Let us denote by $N_{\mathrm{UF}|1}$ the number of codewords that cannot be recovered owing to unrecoverable failures, referred to hereafter as *corrupted codewords*. Owing to the independence of symbol errors, codewords are independently corrupted. Therefore, $N_{\mathrm{UF}|1}$ is binomially distributed with parameter $1 - q_1$, such that

$$E(N_{\mathrm{UF}|1}) = C\,(1 - q_1) \,. \tag{18}$$

Remark 1: For very small values of P_s, it follows from (17) that

$$P_{\mathrm{UF}|1} \approx \begin{cases} (m-1)\, C\, P_s \,, & \text{for } P_s \ll P_s^{(2)} \\ 1 \,, & \text{for } P_s \gg P_s^{(2)} \,. \end{cases} \tag{19}$$

where $P_s^{(2)}$ is obtained from the approximation (19)

$$P_{\mathrm{UF}|1} \approx (m-1)\, C P_s^{(2)} = 1 \tag{20}$$

as follows:

$$\boxed{P_s^{(2)} \triangleq \frac{1}{C} \cdot \frac{1}{m-1}} \,. \tag{21}$$

Note also that from (15) and (18), it follows that

$$E(N_{\mathrm{UF}|1}) \approx C\,(m-1)\, P_s \,, \quad \text{for } P_s \ll \frac{1}{m-1} \,. \tag{22}$$

In particular, for $P_s = P_s^{(2)}$, it holds that $E(N_{\mathrm{UF}|1}) \approx 1$ and this, combined with the fact that $P_{\mathrm{UF}|1} \approx 1$, implies that one of the C codewords is almost surely corrupted.

The expected number $E(N_{\mathrm{UF}|1} | N_{\mathrm{UF}|1} \geq 1)$ of the number of corrupted codewords, given that such codewords exist, is

$$E(N_{\mathrm{UF}|1} | N_{\mathrm{UF}|1} \geq 1) = \frac{E(N_{\mathrm{UF}|1})}{P(N_{\mathrm{UF}|1} \geq 1)} = \frac{E(N_{\mathrm{UF}|1})}{P_{\mathrm{UF}|1}} \tag{23}$$

$$\overset{(19)(22)}{\approx} \begin{cases} 1 \,, & \text{for } P_s \ll P_s^{(2)} \\ C\,(m-1)\, P_s \,, & \text{for } P_s^{(2)} \ll P_s \ll \frac{1}{m-1} \,. \end{cases} \tag{24}$$

When $I_1 > 0$, the number L_1 of lost symbols is $I_1 + 1$. Consequently, the expected number $E(L_1)$ of lost symbols is

$$E(L_1) = \sum_{i=1}^{m-1} (i+1)\, P(I_1 = i) = E(I_1) + 1 - P(I_1 = 0) \,, \tag{25}$$

which using (12), (13), and (14) yields

$$E(L_1) = 1 - q_1 + (m-1)P_s = 1 - (1-P_s)^{m-1} + (m-1)P_s \,. \tag{26}$$

Remark 2: For small values of P_s, it follows from (15) that

$$E(L_1) \approx 2\,(m-1)\, P_s \,, \quad \text{for } P_s \ll \frac{1}{m-1} \,. \tag{27}$$

In particular, the expected number $E(L_1 | L_1 > 0)$ of lost symbols, given that the codeword is corrupted, is

$$E(L_1 | L_1 > 0) = \frac{E(L_1)}{P(L_1 > 0)} = \frac{E(L_1)}{1 - P(L_1 = 0)} \overset{(14)}{=} \frac{E(L_1)}{1 - q_1}$$

$$\overset{(15)(27)}{\approx} 2 \,, \quad \text{for } P_s \ll \frac{1}{m-1} \,. \tag{28}$$

A subsequent (second) device failure may occur during the rebuild process, which is triggered by the initial device failure. The probability $P_{\mathrm{DF}_2|R}$ of data loss due to two device failures, that is, the probability that one of the $m-1$ remaining devices in the array fails during the rebuild process, depends on the duration of the corresponding rebuild time R and the aggregate failure rate of these $m - 1$ highly reliable devices, and is determined as follows [25]:

$$P_{\mathrm{DF}_2|R} \approx (m-1)\, \lambda\, R \,. \tag{29}$$

In particular, it was shown in [29, Lemma 2] that, for highly reliable devices satisfying conditions (5) and (6), the fraction of the rebuild time R still remaining when another device fails is approximately uniformly distributed between 0 and 1. This implies that the probability $P(j|\mathrm{DF}_2, R)$ that the second device

failure occurs during reconstruction of the jth $(1 \le j \le C)$ codeword does not depend on the rebuild duration R and is

$$P(j|\text{DF}_2, R) = \frac{1}{C}, \quad \text{for } j = 1, 2, \ldots, C, \qquad (30)$$

such that for large C, we have

$$E(J) = \sum_{j=1}^{C} j \, P(j|\text{DF}_2, R) = \sum_{j=1}^{C} j \, \frac{1}{C} = \frac{C+1}{2} \approx \frac{C}{2}, \qquad (31)$$

which implies that when the second failure occurs, on average half the codewords have already been considered for reconstruction.

The probability $P_{\text{DF}_2}(j|R)$ that a device failure occurs during reconstruction of the jth $(1 \le j \le C)$ codeword, and given a rebuild time of R, is equal to the product of $P(j|\text{DF}_2, R)$ and $P_{\text{DF}_2|R}$, which, using (29) and (30), yields

$$P_{\text{DF}_2}(j|R) \approx \frac{(m-1)\,\lambda\,R}{C}, \quad \text{for } j = 1, 2, \ldots, C. \qquad (32)$$

The probability P_{DF_2} of data loss due to two device failures, that is, the probability of a device failure during the rebuild process, is obtained by unconditioning (29) on R, that is,

$$P_{\text{DF}_2} = E(P_{\text{DF}_2|R}) \approx (m-1)\,\lambda\,E(R) \stackrel{(4)}{=} (m-1)\,\frac{\lambda}{\mu}. \qquad (33)$$

Consequently, the probability $P_{\text{DF},1}$ of one device failure, that is, the probability of no subsequent device failure during the rebuild, is

$$P_{\text{DF},1} = 1 - P_{\text{DF}_2} \stackrel{(33)}{\approx} 1 - (m-1)\,\frac{\lambda}{\mu}. \qquad (34)$$

Similarly, the probability $P_{\text{DF}_2}(j)$ of a subsequent device failure during reconstruction of the jth $(1 \le j \le C)$ codeword is obtained by unconditioning (32) on R, that is,

$$P_{\text{DF}_2}(j) = E(P_{\text{DF}_2}(j|R)) \approx \frac{(m-1)\,\lambda\,E(R)}{C} \stackrel{(4)}{=} \frac{(m-1)}{C}\,\frac{\lambda}{\mu}. \qquad (35)$$

The probability $P_{\text{UF},1}$ of data loss due to a unrecoverable failures during rebuild in the case of one device failure is obtained by unconditioning the probability $P_{\text{UF}|1}$ of unrecoverable failure during rebuild given one device failure, that is,

$$P_{\text{UF},1} = P_{\text{UF}|1}\,P_{\text{DF},1} \stackrel{(34)}{=} (1 - P_{\text{DF}_2})\,P_{\text{UF}|1} \qquad (36)$$

$$\stackrel{(16)(33)}{\approx} \left[1 - (m-1)\,\frac{\lambda}{\mu}\right](1 - q_1^C), \qquad (37)$$

where q_1 is determined by (14).

Corollary 1: It holds that

$$P_{\text{UF},1} \approx P_{\text{UF}|1} \stackrel{(16))}{=} 1 - q_1^C. \qquad (38)$$

Proof: Immediate from (36) given that, according to (5), it holds that $P_{\text{DF},1} = 1 - (m-1)\,\lambda/\mu \approx 1$ for small values of λ/μ. ∎

Corollary 2: It holds that

$$P_{\text{UF},1} \approx \begin{cases} P_{\text{DF},1}\,(m-1)\,C\,P_s, & \text{for } P_s \ll P_s^{(2)} \\ P_{\text{DF},1}, & \text{for } P_s \gg P_s^{(2)}, \end{cases} \qquad (39)$$

where $P_{\text{DF},1}$ and $P_s^{(2)}$ are given by (34) and (21), respectively.

Proof: Immediate by substituting (19) into (36). ∎

Substituting (17) into (37) yields

$$P_{\text{UF},1} \approx \left[1 - (m-1)\,\frac{\lambda}{\mu}\right]\left[1 - (1 - P_s)^{(m-1)\,C}\right]. \qquad (40)$$

We now proceed to assess the amount of data loss. The expected number $E(S_{\text{U}|1})$ of symbols lost due to unrecoverable failures during rebuild, given one device failure, is

$$E(S_{\text{U}|1}) = C\,E(L_1). \qquad (41)$$

Substituting (26) into (41) yields

$$E(S_{\text{U}|1}) = C\left[1 - (1 - P_s)^{m-1} + (m-1)P_s\right]. \qquad (42)$$

Remark 3: For small values of P_s, $E(L_1)$ is approximated by (27). Consequently, it follows from (41) that

$$E(S_{\text{U}|1}) \approx 2\,C\,(m-1)\,P_s, \quad \text{for } P_s \ll \frac{1}{m-1}. \qquad (43)$$

The expected number $E(S_{\text{U},1})$ of symbols lost due to unrecoverable failures during rebuild in conjunction with one device failure is obtained by unconditioning the conditional expected number $E(S_{\text{U}|1})$ of symbols lost due to unrecoverable failures during rebuild given one device failure, that is,

$$E(S_{\text{U},1}) = E(S_{\text{U}|1})\,P_{\text{DF},1} \stackrel{(34)}{=} (1 - P_{\text{DF}_2})\,E(S_{\text{U}|1}). \qquad (44)$$

Corollary 3: It holds that

$$E(S_{\text{U},1}) \approx E(S_{\text{U}|1}) = C\,E(L_1) \qquad (45)$$

$$\stackrel{(27)}{\approx} 2\,C\,(m-1)\,P_s, \quad \text{for } P_s \ll \frac{1}{m-1}. \qquad (46)$$

Proof: Immediate from (44) given that, according to (5), it holds that $P_{\text{DF},1} = 1 - (m-1)\,\lambda/\mu \approx 1$ for small values of λ/μ, and also using (41). ∎

Substituting (34) and (41) into (44) yields

$$E(S_{\text{U},1}) = C\,E(L_1)\left[1 - (m-1)\,\frac{\lambda}{\mu}\right] \qquad (47)$$

$$\stackrel{(26)}{\approx} C\left[1 - (1 - P_s)^{m-1} + (m-1)P_s\right]\left[1 - (m-1)\,\frac{\lambda}{\mu}\right]. \qquad (48)$$

B. Two Device Failures

As discussed in Section V-A, a subsequent (second) device failure may occur during the rebuild process, which is triggered by the first device failure. In particular, such a failure may occur during reconstruction of the jth $(1 \le j \le C)$ codeword with a probability of $P_{\text{DF}_2}(j)$ determined by (35). This device failure divides the C codewords into two sets:

$S_{1,j}$: the set of $j-1$ codewords already considered for reconstruction, and

$S_{2,j}$: the set of remaining $C-j+1$ codewords, none of which can be reconstructed.

Note that the occurrence of the second device failure does not exclude the possibility of unrecoverable failures being

encountered prior to its occurrence. More specifically, the probability that no unrecoverable failures occur in $S_{1,j}$ is q_1^{j-1}, which implies that the probability $P_{\text{UF in } S_{1,j}|2}$ that one or more unrecoverable failures occur in $S_{1,j}$ is

$$P_{\text{UF in } S_{1,j}|2} = 1 - q_1^{j-1} . \tag{49}$$

Furthermore, such unrecoverable failures entail loss of user data. Let us denote by $N_{\text{UF in } S_{1,j}|2}$ the number of such corrupted codewords. Then it holds that

$$E(N_{\text{UF in } S_{1,j}|2}) = (j-1)(1-q_1) . \tag{50}$$

Also, each of the $C - j + 1$ codewords in $S_{2,j}$ can no longer be reconstructed. In particular, the two symbols of the jth codeword that are stored on the two failed devices can no longer be recovered and are lost. Furthermore, any of its remaining $m - 2$ symbols encountered with unrecoverable errors will also be lost, leading to unrecoverable failure. Thus, the probability p_2 of not encountering an unrecoverable failure due to unrecoverable errors in the remaining $m - 2$ symbols is

$$p_2 = (1 - P_s)^{m-2} , \tag{51}$$

which for very small values of P_s implies that

$$p_2 \approx \begin{cases} 1 - (m-2) P_s , & \text{for } P_s \ll \frac{1}{m-2} \\ 0 , & \text{for } P_s \gg \frac{1}{m-2} . \end{cases} \tag{52}$$

The same applies for the remaining $C - j$ codewords. Therefore the probability that no unrecoverable failures occur in $S_{2,j}$ is p_2^{C-j+1}, which implies that the probability $P_{\text{UF in } S_{2,j}|2}$ that an unrecoverable failure occurs in $S_{2,j}$ is

$$P_{\text{UF in } S_{2,j}|2} = 1 - p_2^{C-j+1} . \tag{53}$$

Furthermore, such unrecoverable failures entail the loss of user data. Let us denote by $N_{\text{UF in } S_{2,j}|2}$ the number of such corrupted codewords. Then it holds that

$$E(N_{\text{UF in } S_{2,j}|2}) = (C-j+1)(1-p_2) . \tag{54}$$

Also, the probability $P_{\text{UF}|2}(j)$ that an unrecoverable failure occurs, given two device failures, is

$$P_{\text{UF}|2}(j) = 1 - q_1^{j-1} p_2^{C-j+1} . \tag{55}$$

The second device failure divides the C codewords into two sets:

S_1 : the set of the codewords already considered for reconstruction when the second device failure occurs, and

S_2 : the set of the remaining codewords, none of which can be reconstructed.

The probability $P_{\text{UF in } S_1|2}$ that an unrecoverable failure occurs in S_1 is obtained by unconditioning (49) on j as follows:

$$P_{\text{UF in } S_1|2} = \sum_{j=1}^{C} P_{\text{UF in } S_{1,j}|2} P(j|\text{DF}_2, R)$$

$$\overset{(30)}{\approx} \sum_{j=1}^{C} (1 - q_1^{j-1}) \frac{1}{C} = 1 - \frac{1 - q_1^C}{C(1-q_1)} , \tag{56}$$

which does not depend on the rebuild duration.

Corollary 4: It holds that

$$P_{\text{UF in } S_1|2} \approx \begin{cases} \dfrac{(C-1)(m-1)}{2} P_s , & \text{for } P_s \ll 2 P_s^{(2)} \\ 1 , & \text{for } P_s \gg 2 P_s^{(2)} , \end{cases} \tag{57}$$

where $P_s^{(2)}$ is determined by (21).

Proof: See Appendix A. ∎

Remark 4: For $P_s \ll P_s^{(2)}$, and considering that C is large, it follows from (19) and (57) that

$$P_{\text{UF in } S_1|2} \approx \frac{1}{2} P_{\text{UF}|1} , \quad \text{for } P_s \ll P_s^{(2)} . \tag{58}$$

This is intuitively obvious because, according to (31), when the second failure occurs, on average half the codewords have already been considered for reconstruction, that is, $E(|S_1|) \approx C/2$.

The expected number $E(N_{\text{UF in } S_1|2})$ of corrupted codewords in S_1 is obtained by unconditioning (54) on j as follows:

$$E(N_{\text{UF in } S_1|2}) = \sum_{j=1}^{C} E(N_{\text{UF in } S_{1,j}|2}) P(j|\text{DF}_2, R)$$

$$\overset{(30)}{\approx} \sum_{j=1}^{C} (j-1)(1-q_1) \frac{1}{C} = \frac{C-1}{2}(1-q_1) \tag{59}$$

$$\overset{(15)}{\approx} \frac{C}{2}(m-1) P_s , \quad \text{for } P_s \ll \frac{1}{m-1} . \tag{60}$$

The probability $P_{\text{UF in } S_2|2}$ that an unrecoverable failure occurs in S_2 is obtained by unconditioning (53) on j as follows:

$$P_{\text{UF in } S_2|2} = \sum_{j=1}^{C} P_{\text{UF in } S_{2,j}|2} P(j|\text{DF}_2, R)$$

$$\approx \sum_{j=1}^{C} (1 - p_2^{C-j+1}) \frac{1}{C} = 1 - \frac{p_2}{C} \frac{1 - p_2^C}{1 - p_2} , \tag{61}$$

which does not depend on the rebuild duration.

Corollary 5: It holds that

$$P_{\text{UF in } S_2|2} \approx \begin{cases} \dfrac{(C+1)(m-2)}{2} P_s , & \text{for } P_s \ll P_{s^*}^{(2)} \\ 1 , & \text{for } P_s \gg P_{s^*}^{(2)} , \end{cases} \tag{62}$$

where $P_{s^*}^{(2)} = 2/[C(m-2)]$. Note that by virtue of (21), it holds that $P_{s^*}^{(2)} = [(m-1)(m-2)] 2 P_s^{(2)} > 2 P_s^{(2)}$.

Proof: See Appendix B. ∎

The probability $P_{\text{UF}|2}$ of data loss due to unrecoverable failures, given two device failures and a rebuild duration of R, is obtained by unconditioning (55) on j and using (30) as

follows:

$$P_{\mathrm{UF}|2} = \sum_{j=1}^{C} P_{\mathrm{UF}|2}(j)\, P(j|\mathrm{DF}_2, R)$$

$$\approx \sum_{j=1}^{C} \left(1 - q_1^{j-1} p_2^{C-j+1} \right) \frac{1}{C} = 1 - \frac{p_2}{C} \frac{p_2^C - q_1^C}{p_2 - q_1}, \tag{63}$$

which does not depend on the rebuild duration.

Corollary 6: It holds that

$$P_{\mathrm{UF}|2} \approx \begin{cases} \left[\dfrac{C-1}{2} + (m-2)\,C \right] P_s, & \text{for } P_s \ll P_s^{(2)} \\ 1, & \text{for } P_s \gg P_s^{(2)}, \end{cases} \tag{64}$$

where $P_s^{(2)}$ is determined by (21).

Proof: See Appendix C. ∎

The probability $P_{\mathrm{UF},2}$ of data loss due to unrecoverable failures in conjunction with two device failures is obtained by unconditioning (63) via (33) as follows:

$$P_{\mathrm{UF},2} = P_{\mathrm{UF}|2}\, P_{\mathrm{DF}_2} \approx \left(1 - \frac{p_2}{C} \frac{p_2^C - q_1^C}{p_2 - q_1} \right) (m-1)\frac{\lambda}{\mu}, \tag{65}$$

where q_1 and p_2 are determined by (14) and (51), respectively.

Corollary 7: It holds that

$$P_{\mathrm{UF},2} \approx \begin{cases} \left[\frac{C-1}{2} + (m-2)\,C \right] (m-1)\frac{\lambda}{\mu} P_s, & \text{for } P_s \ll P_s^{(2)} \\ P_{\mathrm{DF}_2} \approx (m-1)\frac{\lambda}{\mu}, & \text{for } P_s \gg P_s^{(2)}, \end{cases} \tag{66}$$

where $P_s^{(2)}$ is determined by (21).

Proof: Immediate by substituting (64) and (33) into (65). ∎

Remark 5: From (39) and (66), it follows that

$$P_{\mathrm{UF},1} \gg P_{\mathrm{UF},2}, \tag{67}$$

because $P_{\mathrm{UF},2}$ is of the order $O(\lambda/\mu)$, which by virtue of (5) is very small, whereas $P_{\mathrm{UF},1}$ is not.

We now proceed to assess the amount of data loss. As discussed above, the two symbols of each of the $C - j + 1$ codewords in $S_{2,j}$ that are stored on the two failed devices can no longer be recovered and are lost. Thus, the total number $S_{\mathrm{D}}(j)$ of symbols in these $C - j + 1$ codewords that are stored on the two failed devices and are therefore lost is

$$S_{\mathrm{D}}(j) = 2\,(C + 1 - j). \tag{68}$$

Also, the expected total number $E(S_{\mathrm{U},2}^+ \,|\, \mathrm{DF}_2 \text{ at } j)$ of symbols stored in these $C - j + 1$ codewords and lost due to unrecoverable failures is

$$E(S_{\mathrm{U},2}^+ \,|\, \mathrm{DF}_2 \text{ at } j) = (C + 1 - j)(m - 2)\,P_s. \tag{69}$$

Furthermore, each of the $j - 1$ codewords in $S_{1,j}$ loses an expected number of $E(L_1)$ symbols. Consequently, the expected total number $E(S_{\mathrm{U},2}^- \,|\, \mathrm{DF}_2 \text{ at } j)$ of symbols stored in these $j - 1$ codewords and lost due to unrecoverable failures is

$$E(S_{\mathrm{U},2}^- \,|\, \mathrm{DF}_2 \text{ at } j) = (j - 1)\,E(L_1). \tag{70}$$

Unconditioning (68), (69), and (70) on the event of a device failure during reconstruction of the jth codeword, and using (35), yields

$$E(S_{\mathrm{D}}) \approx \sum_{j=1}^{C} 2\,(C + 1 - j)\,\frac{(m-1)}{C}\frac{\lambda}{\mu} \tag{71}$$

$$= (C + 1)\,(m - 1)\frac{\lambda}{\mu}, \tag{72}$$

$$E(S_{\mathrm{U},2}^+) \approx \sum_{j=1}^{C}(C + 1 - j)\,(m - 2)\,P_s\,\frac{(m-1)}{C}\frac{\lambda}{\mu} \tag{73}$$

$$= \frac{C + 1}{2}\,(m - 1)\,(m - 2)\frac{\lambda}{\mu}\,P_s, \tag{74}$$

and

$$E(S_{\mathrm{U},2}^-) \approx \sum_{j=1}^{C}(j - 1)\,E(L_1)\,\frac{(m-1)}{C}\frac{\lambda}{\mu} \tag{75}$$

$$\overset{(26)}{=} \frac{C - 1}{2}\,[1 - (1 - P_s)^{m-1} + (m-1)P_s](m-1)\frac{\lambda}{\mu} \tag{76}$$

$$\overset{(27)}{\approx} (C - 1)\,(m - 1)^2\frac{\lambda}{\mu}\,P_s, \quad \text{for } P_s \ll \frac{1}{m-1}. \tag{77}$$

In particular, from (72) and considering that, for large values of C, $C + 1 \approx C$, it holds that

$$E(S_{\mathrm{D}}) \approx C\,(m - 1)\frac{\lambda}{\mu}. \tag{78}$$

The expected total number $E(S_{\mathrm{U},2})$ of symbols lost due to unrecoverable errors in conjunction with two device failures is

$$E(S_{\mathrm{U},2}) = E(S_{\mathrm{U},2}^+) + E(S_{\mathrm{U},2}^-), \tag{79}$$

where $E(S_{\mathrm{U},2}^+)$ and $E(S_{\mathrm{U},2}^-)$ are determined by (74) and (76), respectively.

Remark 6: From (48), (74), and (76), it follows that $E(S_{\mathrm{U},1}) \gg E(S_{\mathrm{U},2}^-) > E(S_{\mathrm{U},2}^+)$. The first inequality follows from the fact that $E(S_{\mathrm{U},2}^-)$ is of the order $O(\lambda/\mu)$, which is very small, whereas $E(S_{\mathrm{U},1})$ is not. The second inequality follows from the fact that, for large values of C, we have $E(S_{\mathrm{U},2}^-)/E(S_{\mathrm{U},2}^+) \approx [1 - (1 - P_s)^{m-1} + (m-1)P_s]/[(m-2)P_s] > 1$. As discussed above, it follows that $E(S_{\mathrm{U},1}) \gg E(S_{\mathrm{U},2})$. Consequently, the symbols lost due to unrecoverable errors are predominately encountered during a rebuild that is completed without experiencing an additional device failure.

From (43), (45), and (78), it follows that

$$E(S_{\mathrm{U},1}) \ll E(S_{\mathrm{D}}) \Leftrightarrow 2\,C\,(m - 1)\,P_s \ll C\,(m - 1)\frac{\lambda}{\mu}$$

$$\Leftrightarrow P_s \ll P_s^{(3)}, \tag{80}$$

where

$$\boxed{P_s^{(3)} \triangleq \frac{1}{2} \cdot \frac{\lambda}{\mu}.} \tag{81}$$

Remark 7: From (72), it follows that the expected number $E(S_{\mathrm{D}})$ of symbols stored on the two failed devices and lost

is of the order $O(\lambda/\mu)$. Note that the above analysis does not exclude the possibility that additional device failures occur during rebuild. However, the corresponding expected number of the additional lost symbols can be ignored because it is of the order $O((\lambda/\mu)^2)$, which is much smaller than $O(\lambda/\mu)$. This is confirmed by (218), which is derived in Section VI-C and obtains the expected number of lost symbols in conjunction with three device failures.

C. Data Loss

Data loss during rebuild may occur because of another (second) device failure or an unrecoverable failure of one or more codewords, or a combination thereof.

Let P_{DL} denote the probability of data loss. Then, the probability $1-P_{\mathrm{DL}}$ of the rebuild being completed successfully is equal to the product of $1 - P_{\mathrm{DF}_2}$, the probability of not encountering a device failure during rebuild, and $1 - P_{\mathrm{UF}|1}$, the probability of not encountering an unrecoverable failure during rebuild, namely, $1 - P_{\mathrm{DL}} = (1 - P_{\mathrm{DF}_2})(1 - P_{\mathrm{UF}|1})$. Consequently,

$$P_{\mathrm{DL}} = P_{\mathrm{DF}_2} + (1 - P_{\mathrm{DF}_2})\, P_{\mathrm{UF}|1} \overset{(36)}{=} P_{\mathrm{DF}_2} + P_{\mathrm{UF},1}\,. \quad (82)$$

This expresses the fact that a data loss during rebuild may occur either because of two device failures or unrecoverable failures in the case of one device failure. These are two mutually exclusive events. Substituting (17) and (34) into (82) yields

$$P_{\mathrm{DL}} \approx (m-1)\frac{\lambda}{\mu} + \left[1 - (m-1)\frac{\lambda}{\mu}\right]\left[1 - (1 - P_s)^{(m-1)\,C}\right]. \quad (83)$$

Corollary 8: For small values of λ/μ, it holds that

$$P_{\mathrm{DL}} \approx P_{\mathrm{DF}_2} + P_{\mathrm{UF}|1} \quad (84)$$

$$\overset{(33)(17)}{\approx} (m-1)\frac{\lambda}{\mu} + 1 - (1 - P_s)^{(m-1)\,C} \quad (85)$$

$$\overset{(33)(19)}{\approx} \begin{cases} (m-1)\left(\frac{\lambda}{\mu} + C\,P_s\right), & \text{for } P_s \ll P_s^{(2)} \\ 1, & \text{for } P_s \gg P_s^{(2)}\,, \end{cases} \quad (86)$$

where $P_s^{(2)}$ is determined by (21).

Proof: Immediate from (82) because $P_{\mathrm{DF}_2} \ll 1$ due to (5). ∎

Remark 8: When P_s increases and approaches 1, the P_{DL} obtained by (84) and (85) approaches $1 + (m-1)\lambda/\mu$ and therefore exceeds 1.

Remark 9: It follows from (19), (33), and (38) that the range $[0, P_s^{(1)})$ of P_s in which the probabilities $P_{\mathrm{UF},1}$ and $P_{\mathrm{UF}|1}$ are much smaller than the probability P_{DF_2} of encountering a device failure during rebuild is obtained by

$$P_{\mathrm{UF}|1} \overset{(38)}{\approx} P_{\mathrm{UF},1} \ll P_{\mathrm{DF}_2}$$

$$\Leftrightarrow (m-1)\,C\,P_s \ll (m-1)\frac{\lambda}{\mu} \Leftrightarrow P_s \ll P_s^{(1)}, \quad (87)$$

where

$$\boxed{P_s^{(1)} \triangleq \frac{1}{C}\cdot\frac{\lambda}{\mu}}\,. \quad (88)$$

Also, it follows from (86) and (87) that

$$P_{\mathrm{DL}} \approx \begin{cases} (m-1)\frac{\lambda}{\mu}, & \text{for } P_s \ll P_s^{(1)} \\ (m-1)\,C\,P_s, & \text{for } P_s^{(1)} \ll P_s \ll P_s^{(2)} \\ 1, & \text{for } P_s \gg P_s^{(2)}\,, \end{cases} \quad (89)$$

where $P_s^{(1)}$ and $P_s^{(2)}$ are determined by (88) and (21), respectively. Note that P_{DL}, as a function of P_s, exhibits two plateaus in the intervals $[0, P_s^{(1)})$ and $(P_s^{(2)}, 1]$, respectively.

Unrecoverable failures may occur in the cases of one device failure and two device failures. Consequently, the probability P_{UF} of encountering one or more unrecoverable failures during rebuild is

$$P_{\mathrm{UF}} = P_{\mathrm{UF},1} + P_{\mathrm{UF},2} \quad (90)$$

$$\overset{(67)}{\approx} P_{\mathrm{UF},1} \overset{(40)}{\approx} \left[1 - (m-1)\frac{\lambda}{\mu}\right]\left[1 - (1 - P_s)^{(m-1)\,C}\right] \quad (91)$$

$$\overset{(38)}{\approx} P_{\mathrm{UF}|1} \overset{(19)}{\approx} \begin{cases} (m-1)\,C\,P_s, & \text{for } P_s \ll P_s^{(2)} \\ 1, & \text{for } P_s \gg P_s^{(2)}\,, \end{cases} \quad (92)$$

where $P_{\mathrm{UF},1}$ and $P_{\mathrm{UF},2}$ are given by (40) and (65), respectively.

D. Amount of Data Loss

As discussed in Section V-C, data loss during rebuild may occur because of another (second) device failure or an unrecoverable failure of one or more codewords, or a combination thereof. Note that in all cases, data loss cannot involve only parity data, but also loss of user data.

Data loss during rebuild may occur because of unrecoverable failures in the cases of one device failure or two device failures. Consequently, the expected number $E(S_{\mathrm{U}})$ of symbols lost due to unrecoverable errors is obtained as follows:

$$E(S_{\mathrm{U}}) = E(S_{\mathrm{U},1}) + E(S_{\mathrm{U},2})\,, \quad (93)$$

where $E(S_{\mathrm{U},1})$ and $E(S_{\mathrm{U},2})$ are determined by (48) and (79), respectively. Moreover, according to Remark 6, it holds that

$$E(S_{\mathrm{U}}) \approx E(S_{\mathrm{U},1}) \quad (94)$$

$$\overset{(46)}{\approx} 2\,C\,(m-1)\,P_s\,, \quad \text{for } P_s \ll \frac{1}{m-1}\,. \quad (95)$$

The expected total number $E(S)$ of lost symbols is

$$E(S) = E(S_{\mathrm{D}}) + E(S_{\mathrm{U}})\,, \quad (96)$$

where $E(S_{\mathrm{D}})$ and $E(S_{\mathrm{U}})$ are determined by (72) and (93), respectively.

Remark 10: It follows from (45), (78), (94), and (96) that

$$E(S) \approx C\left[(m-1)\frac{\lambda}{\mu} + E(L_1)\right] \quad (97)$$

$$\overset{(27)}{\approx} C\,(m-1)\left(\frac{\lambda}{\mu} + 2\,P_s\right), \text{ for } P_s \ll \frac{1}{m-1}\,, \quad (98)$$

where $E(L_1)$ is determined by (26). In particular, for $P_s = 0$, it holds that $E(S) = E(S_{\mathrm{D}}) \approx C\,(m-1)\,\lambda/\mu$.

Remark 11: When P_s increases and approaches 1, it follows from (48), (72), (74), (76), (93), and (96) that $E(S)$ approaches $C\,m$. This is intuitively obvious because when

$P_s = 1$, all $C\,m$ symbols stored in the the RAID-5 array are lost owing to unrecoverable errors.

We now proceed to derive $E(Q)$, the expected amount of lost user data. First, we note that the expected number of lost user symbols is equal to the product of the storage efficiency and the expected number of lost symbols. Consequently, it follows from (1) that

$$E(Q) = \frac{l}{m}\,E(S)\,s \overset{(3)}{=} \frac{l}{m}\,\frac{E(S)}{C}\,c\,, \qquad (99)$$

where $E(S)$ is given by (96) and s denotes the symbol size.

Similar expressions for the expected amounts $E(Q_{\mathrm{DF_2}})$ and $E(Q_{\mathrm{UF}})$ of user data lost due to device and unrecoverable failures are obtained from $E(S_{\mathrm{D}})$ and $E(S_{\mathrm{U}})$, respectively, as follows:

$$E(Q_{\mathrm{DF_2}}) = \frac{l}{m}\,E(S_{\mathrm{D}})\,s \overset{(3)}{=} \frac{l}{m}\,\frac{E(S_{\mathrm{D}})}{C}\,c\,, \qquad (100)$$

$$\overset{(78)}{\approx} \frac{l}{m}\,(m-1)\,\frac{\lambda}{\mu}\,c \qquad (101)$$

and

$$E(Q_{\mathrm{UF}}) = \frac{l}{m}\,E(S_{\mathrm{U}})\,s \overset{(3)}{=} \frac{l}{m}\,\frac{E(S_{\mathrm{U}})}{C}\,c \qquad (102)$$

$$\overset{(95)}{\approx} 2\,\frac{l}{m}\,(m-1)\,c\,P_s\,, \quad \text{for } P_s \ll \frac{1}{m-1}\,, \quad (103)$$

where $E(S_{\mathrm{D}})$ and $E(S_{\mathrm{U}})$ are determined by (72) and (93), respectively.

Substituting (97) and (98) into (99) yields

$$E(Q) \approx \frac{l}{m}\,\left[(m-1)\,\frac{\lambda}{\mu} + E(L_1)\right]c \qquad (104)$$

$$\approx \frac{l}{m}\,(m-1)\,\left(\frac{\lambda}{\mu} + 2\,P_s\right)c,\ \text{for } P_s \ll \frac{1}{m-1}, \qquad (105)$$

where $E(L_1)$ is given by (26). In particular, for $P_s = 0$, it holds that $E(Q) = E(Q_{\mathrm{DF_2}})$, which is determined by (101).

From (96), (99), (100), and (102), it holds that

$$E(Q) = E(Q_{\mathrm{DF_2}}) + E(Q_{\mathrm{UF}})\,. \qquad (106)$$

Also, the expected amounts $E(Q_{\mathrm{UF},1})$ and $E(Q_{\mathrm{UF},2})$ of user data lost due to unrecoverable failures in the cases of one device failure and two device failures are

$$E(Q_{\mathrm{UF},1}) = \frac{l}{m}\,\frac{E(S_{\mathrm{U},1})}{C}\,c \qquad (107)$$

and

$$E(Q_{\mathrm{UF},2}) = \frac{l}{m}\,\frac{E(S_{\mathrm{U},2})}{C}\,c\,, \qquad (108)$$

where $E(S_{\mathrm{U},1})$ and $E(S_{\mathrm{U},2})$ are determined by (48) and (79), respectively.

Remark 12: From (94), (102), and (107), it follows that

$$E(Q_{\mathrm{UF}}) \approx E(Q_{\mathrm{UF},1})\,. \qquad (109)$$

Remark 13: From (80), (94), (100), and (102), it follows that

$$E(Q_{\mathrm{UF}}) \ll E(Q_{\mathrm{DF_2}}) \Leftrightarrow P_s \ll P_s^{(3)}\,, \qquad (110)$$

where $P_s^{(3)}$ is determined by (81).

Also, from (101), (103), (106), and (110), it follows that

$$E(Q) \approx \begin{cases} E(Q_{\mathrm{DF_2}})\,, & \text{for } P_s \ll P_s^{(3)} \\ E(Q_{\mathrm{UF}})\,, & \text{for } P_s \gg P_s^{(3)} \end{cases} \qquad (111)$$

$$\approx \begin{cases} \frac{l}{m}\,(m-1)\,\frac{\lambda}{\mu}\,c\,, & \text{for } P_s \ll P_s^{(3)} \\ 2\,\frac{l}{m}\,(m-1)\,c\,P_s\,, & \text{for } P_s^{(3)} \ll P_s \ll \frac{1}{m-1}\,. \end{cases} \qquad (112)$$

Remark 14: When P_s increases and approaches 1, from (99) and according to Remark 11, it follows that $E(Q)$ approaches $c\,l$. This is intuitively obvious because when $P_s = 1$, upon the first device failure, the entire amount $c\,l$ of user data stored in the RAID-5 array is lost owing to unrecoverable errors.

E. Reliability Metrics

The MTTDL normalized to $1/\lambda$ is obtained by substituting (83) into (9) as follows:

$$\lambda\,\mathrm{MTTDL} \approx$$
$$\frac{1}{n\left\{(m-1)\,\frac{\lambda}{\mu} + \left[1 - (m-1)\,\frac{\lambda}{\mu}\right]\left[1 - (1-P_s)^{(m-1)\,C}\right]\right\}}\,, \qquad (113)$$

where C and λ/μ are determined by (3) and (5), respectively. In particular, substituting (86) into (9) yields

$$\lambda\,\mathrm{MTTDL} \approx \begin{cases} \dfrac{1}{n\,(m-1)\,\left(\frac{\lambda}{\mu} + C\,P_s\right)}, & \text{for } P_s \ll P_s^{(2)} \\[2mm] \dfrac{1}{n}\,, & \text{for } P_s \gg P_s^{(2)}\,. \end{cases} \qquad (114)$$

Note that MTTDL is insensitive to device failure and rebuild time distributions; it depends only on their means $1/\lambda$ and $1/\mu$, respectively. In particular, the normalized MTTDL depends only on the ratio λ/μ of their means. Also, for $\lambda/\mu \ll 1$ and $n = m = N$, the MTTDL derived in (113) is approximately equal to

$$\mathrm{MTTDL} \approx \frac{1 + (2N-1)\,\frac{\lambda}{\mu}}{N\,\lambda\left\{(N-1)\,\frac{\lambda}{\mu} + \left[1 - (1-P_s)^{(N-1)\,C}\right]\right\}}\,, \qquad (115)$$

which is Equation (43) derived in [21]. Furthermore, for $P_s = 0$ and $n = m = N$, (113) yields

$$\mathrm{MTTDL} \approx \frac{\mu}{N\,(N-1)\,\lambda^2}\,, \qquad (116)$$

which is the same result as the one derived in [2, 3] (for a single array, namely, $n_G = 1$).

The EAFDL is obtained by substituting (99) into (10). In particular, the EAFDL normalized to λ is obtained by substituting (105) into (10) as follows:

$$\mathrm{EAFDL}/\lambda \approx (m-1)\,\left(\frac{\lambda}{\mu} + 2\,P_s\right)\,, \quad \text{for } P_s \ll \frac{1}{m-1}\,. \qquad (117)$$

where λ/μ is determined by (5). Note that EAFDL is insensitive to device failure and rebuild time distributions; it depends only on their means $1/\lambda$ and $1/\mu$, respectively. In particular,

the normalized EAFDL only depends on the ratio λ/μ of their means. Also, for $P_s = 0$, (117) is in agreement with Equation (74) of [14] (with $c/b = 1/\mu$ and $\phi = 1$).

The value of $E(H)$ is obtained by substituting (83) and (99) into (11). In particular, the $E(H)$ normalized to c is obtained by substituting (83) and (105) into (11) as follows:

$$
E(H)/c \approx
$$

$$
\frac{\frac{l}{m}(m-1)\left(\frac{\lambda}{\mu} + 2P_s\right)}{(m-1)\frac{\lambda}{\mu} + \left[1 - (m-1)\frac{\lambda}{\mu}\right]\left[1 - (1-P_s)^{(m-1)C}\right]},
$$

$$
\text{for } P_s \ll \frac{1}{m-1}, \quad (118)
$$

where C and λ/μ are determined by (3) and (5), respectively.

Note that $E(H)$ does not depend on the device failure nor the rebuild time distributions; it only depends on the ratio of their means λ/μ. Also, for $P_s = 0$, (118) yields $E(H)/c = l/m$, which is in agreement with Equation (75) of [14].

Similar to (11), the expected amounts $E(H_{\mathrm{DF}_2})$ and $E(H_{\mathrm{UF}})$ of user data lost due to device and unrecoverable failures, given that such failures have occurred, are

$$
E(H_{\mathrm{DF}_2}) = \frac{E(Q_{\mathrm{DF}_2})}{P_{\mathrm{DF}_2}}, \quad \text{and} \quad E(H_{\mathrm{UF}}) = \frac{E(Q_{\mathrm{UF}})}{P_{\mathrm{UF}}}, \quad (119)
$$

respectively.

From (11), (106), and (119), we deduce that the following relation holds

$$
E(H) = \frac{P_{\mathrm{DF}_2}}{P_{\mathrm{DL}}} E(H_{\mathrm{DF}_2}) + \frac{P_{\mathrm{UF}}}{P_{\mathrm{DL}}} E(H_{\mathrm{UF}}). \quad (120)
$$

Note that this is not a weighted average of $E(H_{\mathrm{DF}_2})$ and $E(H_{\mathrm{UF}})$ because a subsequent device failure and unrecoverable failures during rebuild are not mutually exclusive, and therefore, and according to (82) and (90), the sum of weights is not equal to but close to 1.

Remark 15: According to (21), (81), and (88), it holds that $P_s^{(1)} \ll \min(P_s^{(2)}, P_s^{(3)})$ and $P_s^{(2)} \lessgtr P_s^{(3)} \Leftrightarrow \lambda/\mu \gtrless 2/[C(m-1)]$. From (89) and (112), it follows that

$$
E(H)/c
$$

$$
\approx \begin{cases}
\frac{l}{m}, & \text{for } P_s \ll P_s^{(1)} \\
\frac{l}{m}\frac{\lambda}{\mu}\frac{1}{C P_s}, & \text{for } P_s^{(1)} \ll P_s \ll P_{\min}^{(2,3)} \\
2\frac{l}{m}\max\left(\frac{m-1}{2}\frac{\lambda}{\mu}, \frac{1}{C}\right), & \text{for } P_{\min}^{(2,3)} \ll P_s \ll P_{\max}^{(2,3)} \\
2\frac{l}{m}(m-1)P_s, & \text{for } P_{\max}^{(2,3)} \ll P_s \ll \frac{1}{m-1},
\end{cases}
$$

$$
(121)
$$

where

$$
P_{\min}^{(2,3)} \triangleq \min(P_s^{(2)}, P_s^{(3)}) \quad \text{and} \quad P_{\max}^{(2,3)} \triangleq \max(P_s^{(2)}, P_s^{(3)}). \quad (122)
$$

Note that $E(H)$, as a function of P_s, exhibits two plateaus in the intervals $[0, P_s^{(1)})$ and $(P_{\min}^{(2,3)}, P_{\max}^{(2,3)})$, respectively.

Substituting (33), (91), (101), and (103) into (119) yields

$$
E(H_{\mathrm{DF}_2})/c \approx \frac{l}{m}, \quad (123)
$$

and

$$
\frac{E(H_{\mathrm{UF}})}{c} \approx \frac{2\frac{l}{m}(m-1)P_s}{\left[1 - (m-1)\frac{\lambda}{\mu}\right]\left[1 - (1-P_s)^{(m-1)C}\right]},
$$

$$
\text{for } P_s \ll \frac{1}{m-1}. \quad (124)
$$

where C and λ/μ are determined by (3) and (5), respectively.

Remark 16: Substituting (92) and (103) into (119) yields

$$
\frac{E(H_{\mathrm{UF}})}{c} \approx \begin{cases}
2\frac{l}{m}\frac{1}{C}, & \text{for } P_s \ll P_s^{(2)} \\
2\frac{l}{m}(m-1)P_s, & \text{for } P_s^{(2)} \ll P_s \ll \frac{1}{m-1}.
\end{cases}
$$

$$
(125)
$$

Remark 17: When P_s increases and approaches 1, it follows from (11), (83), and Remark 14 that $E(H)$ approaches cl. This is intuitively obvious because when $P_s = 1$, the entire amount cl of user data stored in the system is lost owing to unrecoverable errors.

VI. RAID-6 SYSTEMS

Here we derive the reliability metrics for a RAID-6 system. When a storage device of a RAID-6 array fails, the C codewords stored in the array lose one of their symbols. Using the direct-path-approximation methodology, we proceed by considering only the subsequent potential data losses and device failures related to the affected array.

A. One Device Failure

The rebuild process attempts to restore the C codewords of the affected array sequentially. Let us consider such a codeword and let L_1 be the number of symbols permanently lost and I_1 be the number of symbols in the codeword with unrecoverable errors. The probability distribution of I_1 is determined by (12). Clearly, the symbol lost due to the device failure can be corrected by the RAID-6 capability only if at least $m-2$ of the remaining $m-1$ symbols can be read. Thus, $L_1 = 0$ if and only if $I_1 \le 1$. Using (12), the probability q_1 that a codeword can be restored is

$$
q_1 = P(L_1 = 0) = P(I_1 \le 1) \stackrel{(12)}{=} [1+(m-2)P_s](1-P_s)^{m-2}, \quad (126)
$$

which for very small values of P_s implies that

$$
q_1 \approx \begin{cases}
1 - \frac{(m-1)(m-2)}{2}P_s^2, & \text{for } P_s \ll \sqrt{\frac{2}{(m-1)(m-2)}} \\
0, & \text{for } P_s \gg \sqrt{\frac{2}{(m-1)(m-2)}}.
\end{cases}
$$

$$
(127)
$$

Note that, if a codeword is corrupted, then at least one of its l user-data symbols is lost. We now deduce that the conditional probability $P_{\mathrm{UF}|1}$ of encountering an unrecoverable failure during the rebuild process of the C codewords in the case of one device failure is

$$
P_{\mathrm{UF}|1} = 1 - q_1^C \stackrel{(126)}{=} 1 - [1+(m-2)P_s]^C(1-P_s)^{(m-2)C}, \quad (128)
$$

Remark 18: For very small values of P_s, q_1 is approximated by (127). Consequently, it follows from (128) that

$$
P_{\mathrm{UF}|1} \approx \begin{cases}
C\frac{(m-1)(m-2)}{2}P_s^2, & \text{for } P_s \ll P_{s*}^{(4)} \\
1, & \text{for } P_s \gg P_{s*}^{(4)}.
\end{cases}
$$

$$
(129)
$$

where $P_{s^*}^{(4)}$ is obtained from the approximation (129)

$$P_{\text{UF}|1} \approx \frac{(m-1)(m-2)}{2} \, C \, P_s^2 = 1 \qquad (130)$$

as follows:

$$P_{s^*}^{(4)} \triangleq \sqrt{\frac{2}{C(m-1)(m-2)}} \, . \qquad (131)$$

Note also that, for $P_s \ll \sqrt{\frac{2}{(m-1)(m-2)}}$ and from (18) and (127), the expected number $E(N_{\text{UF}|1})$ of corrupted codewords is

$$E(N_{\text{UF}|1}) \approx C \, \frac{(m-1)(m-2)}{2} \, P_s^2 \, . \qquad (132)$$

In particular, for $P_s = P_{s^*}^{(4)}$, it holds that $E(N_{\text{UF}|1}) \approx 1$ and this, combined with the fact that $P_{\text{UF}|1} \approx 1$, implies that one of the C codewords is almost surely corrupted.

The expected number $E(N_{\text{UF}|1}|N_{\text{UF}|1} \geq 1)$ of corrupted codewords, given that such codewords exist, is derived using (23), (129), and (132) as follows:

$$E(N_{\text{UF}|1}|N_{\text{UF}|1} \geq 1) = \frac{E(N_{\text{UF}|1})}{P(N_{\text{UF}|1} \geq 1)} = \frac{E(N_{\text{UF}|1})}{P_{\text{UF}|1}}$$

$$\approx \begin{cases} 1 \, , & \text{for } P_s \ll P_{s^*}^{(4)} \\ C \, \frac{(m-1)(m-2)}{2} \, P_s^2 \, , & \text{for } P_{s^*}^{(4)} \ll P_s \ll \sqrt{\frac{2}{(m-1)(m-2)}} \, . \end{cases} \qquad (133)$$

When $I_1 > 1$, the number L_1 of lost symbols is $I_1 + 1$. Consequently, the expected number $E(L_1)$ of lost symbols is

$$E(L_1) = \sum_{i=2}^{m-1} (i+1) \, P(I_1 = i)$$

$$= E(I_1) + 1 - P(I_1 = 0) - 2 \, P(I_1 = 1)$$

$$\overset{(12)(13)}{=} 1 - (1-P_s)^{m-1} + (m-1) \, P_s \, [1 - 2(1-P_s)^{m-2}] \, . \qquad (134)$$

Remark 19: From (134), it follows that

$$E(L_1) \approx \frac{3}{2} \, (m-1)(m-2) \, P_s^2 \, , \quad \text{for } P_s \ll \frac{1}{m-1} \, . \qquad (135)$$

In particular, the expected number $E(L_1|L_1 > 0)$ of lost symbols, given that the codeword cannot be restored, is

$$E(L_1|L_1 > 0) = \frac{E(L_1)}{P(L_1 > 0)} = \frac{E(L_1)}{1 - P(L_1 = 0)} \overset{(126)}{=} \frac{E(L_1)}{1 - q_1}$$

$$\overset{(127)(135)}{\approx} 3 \, , \qquad \text{for } P_s \ll \frac{1}{m-1} \, . \qquad (136)$$

The probability $P_{\text{UF},1}$ of data loss due to unrecoverable failures during rebuild in the case of one device failure is obtained from (37), that is,

$$P_{\text{UF},1} \approx \left[1 - (m-1) \, \frac{\lambda}{\mu} \right] (1 - q_1^C) \, , \qquad (137)$$

where q_1 in the case of RAID-6 is determined by (126).

Corollary 9: It holds that

$$P_{\text{UF},1} \approx \begin{cases} P_{\text{DF},1} \, \dfrac{(m-1)(m-2)}{2} \, C \, P_s^2 \, , & \text{for } P_s \ll P_{s^*}^{(4)} \\[2mm] P_{\text{DF},1} \, , & \text{for } P_s \gg P_{s^*}^{(4)} \, , \end{cases} \qquad (138)$$

where $P_{\text{DF},1}$ and $P_{s^*}^{(4)}$ are given by (34) and (131), respectively.

Proof: Immediate by substituting (129) into (36). ∎

We now proceed to assess the amount of data loss. The expected number $E(S_{\text{U}|1})$ of symbols lost due to unrecoverable failures during rebuild, given one device failure, is determined by (41). Substituting (134) into (41) yields

$$E(S_{\text{U}|1}) = C \, \{ 1 - (1-P_s)^{m-1} + (m-1) \, P_s \, [1 - 2(1-P_s)^{m-2}] \} \, . \qquad (139)$$

Remark 20: For small values of P_s, $E(L_1)$ is approximated by (135). Consequently, it follows from (41) that

$$E(S_{\text{U}|1}) \approx \frac{3}{2} \, C \, (m-1)(m-2) \, P_s^2 \, , \quad \text{for } P_s \ll \frac{1}{m-1} \, . \qquad (140)$$

The expected number $E(S_{\text{U},1})$ of symbols lost due to unrecoverable failures during rebuild, in the case of one device failure during rebuild, is determined by (44), and consequently by (47), that is,

$$E(S_{\text{U},1}) \approx C \, E(L_1) \left[1 - (m-1) \, \frac{\lambda}{\mu} \right] \, , \qquad (141)$$

where $E(L_1)$ is determined by (134).

Remark 21: The relations given in (38) and (45), that is,

$$P_{\text{UF},1} \approx P_{\text{UF}|1} = 1 - q_1^C \qquad (142)$$

and

$$E(S_{\text{U},1}) \approx E(S_{\text{U}|1}) = C \, E(L_1) \, , \qquad (143)$$

hold for both RAID-5 and RAID-6 redundancy schemes, where q_1 is given by (14) and (126), respectively, and $E(L_1)$ by (26) and (134), respectively. In fact, these two relations are general because they apply to any redundancy scheme with the corresponding q_1 and $E(L_1)$ measures evaluated accordingly.

Remark 22: From (135) and (143), it follows that

$$E(S_{\text{U},1}) \approx \frac{3}{2} \, C \, (m-1)(m-2) \, P_s^2 \, , \quad \text{for } P_s \ll \frac{1}{m-1} \, . \qquad (144)$$

B. Two Device Failures

As discussed in Section V-B, a subsequent (second) device failure may occur during the rebuild process, which is triggered by the first device failure. In particular, such a failure may occur during reconstruction of the jth ($1 \leq j \leq C$) codeword with a probability of $P_{\text{DF}_2}(j)$ determined by (35).

The rebuild process attempts to restore the jth codeword as well as the remaining $C - j$ codewords of the affected array sequentially. Let us consider such a codeword and let L_2 be the number of symbols permanently lost and I_2 be the number

of symbols in the codeword with unrecoverable errors. Clearly, I_2 is binomially distributed with parameter P_s, that is

$$P(I_2 = i) = \binom{m-2}{i} P_s^i (1-P_s)^{m-2-i}, \text{ for } i = 0, \ldots, m-2 \,,$$
(145)

such that

$$E(I_2) = \sum_{i=1}^{m-2} i \, P(I_2 = i) \; = \; (m-2) \, P_s \,.$$
(146)

The symbols lost due to the two device failures can be corrected by the RAID-6 capability only if the remaining $m-2$ symbols can be read. Thus, $L_2 = 0$ if and only if $I_2 = 0$. Using (145), the probability q_2 that a codeword can be restored is

$$q_2 = P(L_2 = 0) = P(I_2 = 0) = (1-P_s)^{m-2} \stackrel{(51)}{=} p_2 \,, \quad (147)$$

which for very small values of P_s implies that

$$q_2 \approx \begin{cases} 1 - (m-2) \, P_s \,, & \text{for } P_s \ll \frac{1}{m-2} \\ 0 \,, & \text{for } P_s \gg \frac{1}{m-2} \,. \end{cases}$$
(148)

When $I_2 > 0$, the number L_2 of lost symbols is $I_1 + 2$. Consequently, the expected number $E(L_2)$ of lost symbols is

$$E(L_2) = \sum_{i=1}^{m-2} (i+2) \, P(I_2 = i)$$

$$= E(I_2) + 2 \sum_{i=1}^{m-2} P(I_2 = i) = E(I_2) + 2 \, (1 - q_2)$$

$$\stackrel{(146)(147)}{=} (m-2) \, P_s + 2 \, [1 - (1 - P_s)^{m-2}] \,.$$
(149)

Remark 23: From (149), it follows that

$$E(L_2) \approx 3 \, (m-2) \, P_s \,, \quad \text{for } P_s \ll \frac{1}{m-2} \,.$$
(150)

In particular, the expected number $E(L_2 | L_2 > 0)$ of lost symbols, given that the codeword cannot be restored, is

$$E(L_2 | L_2 > 0) = \frac{E(L_2)}{P(L_2 > 0)} = \frac{E(L_2)}{1 - P(L_2 = 0)} \stackrel{(147)}{=} \frac{E(L_2)}{1 - q_2}$$

$$\stackrel{(127)(135)}{\approx} 3 \,, \quad \text{for } P_s \ll \frac{1}{m-1} \,.$$
(151)

As discussed in Section V-B, the C codewords are divided into two sets: $S_{1,j}$, the set of $j-1$ codewords already considered for reconstruction, and $S_{2,j}$, the set of the remaining $C-j+1$ codewords. In contrast to RAID-5, the codewords in $S_{2,j}$ could be reconstructed by the RAID-6 capability. Therefore, the probability that no unrecoverable failures occur in $S_{2,j}$ is q_2^{C-j+1}, which implies that the probability $P_{\text{UF in } S_{2,j}|2}$ that an unrecoverable failure occurs in $S_{2,j}$ is

$$P_{\text{UF in } S_{2,j}|2} = 1 - q_2^{C-j+1} \,.$$
(152)

Furthermore, such unrecoverable failures entail the loss of user data. Let us denote by $N_{\text{UF in } S_{2,j}|2}$ the number of such corrupted codewords. Then it holds that

$$E(N_{\text{UF in } S_{2,j}|2}) = (C - j + 1) \, (1 - q_2) \,.$$
(153)

Also, the probability $P_{\text{UF}|2}(j)$ that an unrecoverable failure occurs, given two device failures, is

$$P_{\text{UF}|2}(j) = 1 - q_1^{j-1} \, q_2^{C-j+1} \,.$$
(154)

As discussed in Section V-B, the second device failure divides the C codewords into two sets: S_1, the set of codewords already considered for reconstruction when the second device failure occurs, and S_2, the set of the remaining codewords. The probability $P_{\text{UF in } S_1|2}$ that an unrecoverable failure occurs in S_1 is given by (56), where q_1 is determined by (126).

Corollary 10: It holds that

$$P_{\text{UF in } S_1|2} \approx \begin{cases} \dfrac{(C-1)(m-1)(m-2)}{4} \, P_s^2 \,, & \text{for } P_s \ll P_{s^*} \\ 1 \,, & \text{for } P_s \gg P_{s^*} \,, \end{cases}$$
(155)

where P_{s^*} is obtained from the approximation (155)

$$P_{\text{UF in } S_1|2} \approx \frac{(C-1)(m-1)(m-2)}{4} \, P_s^2 = 1 \quad (156)$$

and considering that C is large, as follows:

$$P_{s^*} = \frac{2}{\sqrt{C(m-1)(m-2)}} \approx P_{s^*}^{(4)} \,.$$
(157)

Proof: See Appendix D. ∎

Remark 24: For $P_s \ll P_{s^*}^{(4)}$, and considering that C is large, it follows from (129) and (155) that

$$P_{\text{UF in } S_1|2} \approx \frac{1}{2} \, P_{\text{UF}|1} \,, \quad \text{for } P_s \ll P_{s^*}^{(4)} \,.$$
(158)

This is intuitively obvious because, according to (31), when the second failure occurs, half the codewords on the average have already been considered for reconstruction, that is, $E(|S_1|) \approx C/2$.

The probability $P_{\text{UF in } S_2|2}$ that an unrecoverable failure occurs in S_2 is obtained by unconditioning (152) on j as follows:

$$P_{\text{UF in } S_2|2} = \sum_{j=1}^{C} P_{\text{UF in } S_{2,j}|2} P(j | \text{DF}_2, R)$$

$$\stackrel{(30)}{\approx} \sum_{j=1}^{C} (1 - q_2^{C-j+1}) \frac{1}{C} = 1 - \frac{q_2}{C} \frac{1 - q_2^C}{1 - q_2} \,, \quad (159)$$

which does not depend on the rebuild duration.

Corollary 11: It holds that

$$P_{\text{UF in } S_2|2} \approx \begin{cases} \dfrac{(C+1)(m-2)}{2} \, P_s \,, & \text{for } P_s \ll P_{s^*}^{(2)} \\ 1 \,, & \text{for } P_s \gg P_{s^*}^{(2)} \,, \end{cases}$$
(160)

where $P_{s^*}^{(2)}$ is obtained from the approximation (160)

$$P_{\text{UF in } S_2|2} \approx \frac{(C+1)(m-2)}{2} \, P_s = 1 \quad (161)$$

and is

$$\boxed{P_{s^*}^{(2)} \triangleq \frac{1}{C} \cdot \frac{2}{m-2} \,.}$$
(162)

Proof: Immediate from Corollary 5 and by recognizing that $P_{\text{UF in } S_2|2}$ derived by (159) is equal to the corresponding measure obtained in the case of RAID-5 as expressed by (61) because, according to (147), $q_2 = p_2$. ∎

Remark 25: From (148) and for $P_s \ll 1/(m-2)$, it follows that $q_2 \approx 1$. Furthermore, $\log(q_2) = -(1-q_2) + O((1-q_2)^2) \approx -(1-q_2) \approx -(1-q_2)/q_2$. Consequently, substituting the term $q_2/(1-q_2)$ on the right-hand side of (159) with $-1/\log(q_2)$ yields

$$P_{\text{UF in } S_2|2} \approx 1 + \frac{1-q_2^C}{C\log(q_2)} = 1 + \frac{1-q_2^C}{\log(q_2^C)}, \quad (163)$$

where q_2 is determined by (147).

The expected number $E(N_{\text{UF in } S_1|2})$ of corrupted codewords in S_1 is determined by (59). Also, the expected number $E(N_{\text{UF in } S_2|2})$ of corrupted codewords in S_2 is obtained by unconditioning (153) on j as follows:

$$E(N_{\text{UF in } S_2|2}) = \sum_{j=1}^{C} E(N_{\text{UF in } S_{2,j}|2}) P(j|\text{DF}_2, R)$$

$$\overset{(30)}{\approx} \sum_{j=1}^{C} (C-j+1)(1-q_2) \frac{1}{C}$$

$$= \frac{C-1}{2}(1-q_2) \approx \frac{C}{2}(1-q_2) \quad (164)$$

$$\overset{(148)}{\approx} \frac{C}{2}(m-2)P_s, \quad \text{for } P_s \ll \frac{1}{m-2}. \quad (165)$$

In particular, for $P_s = P_{s*}^{(2)}$ as determined by (162), it holds that $E(N_{\text{UF in } S_2|2}) \approx 1$ and this, combined with the fact that $P_{\text{UF in } S_2|2} \approx 1$ as derived by (160), implies that one of the codewords in S_2 is almost surely corrupted. Also, (127) implies that $1-q_1 = 2(m-1)/[(m-2)C^2]$. Thus, from (59), it follows that $E(N_{\text{UF in } S_1|2}) \approx (m-1)/[(m-2)C]$, which is negligible compared with $E(N_{\text{UF in } S_2|2})$.

The expected number $E(N_{\text{UF in } S_2|2}|N_{\text{UF in } S_2|2} \geq 1)$ of corrupted codewords in S_2, given that such codewords exist, is

$$E(N_{\text{UF in } S_2|2}|N_{\text{UF in } S_2|2} \geq 1) = \frac{E(N_{\text{UF in } S_2|2})}{P_{\text{UF in } S_2|2}} \quad (166)$$

$$\overset{(160)(165)}{\approx} \begin{cases} 1, & \text{for } P_s \ll P_{s*}^{(2)} \\ \frac{C}{2}(m-2)P_s, & \text{for } P_{s*}^{(2)} \ll P_s \ll \frac{1}{m-1}. \end{cases} \quad (167)$$

The probability $P_{\text{UF}|2}$ of a data loss due to unrecoverable failures given two device failures and a rebuild duration of R is obtained by unconditioning (154) on j and using (30) as follows:

$$P_{\text{UF}|2} = \sum_{j=1}^{C} P_{\text{UF}|2}(j) P(j|\text{DF}_2, R)$$

$$\approx \sum_{j=1}^{C} \left(1 - q_1^{j-1} q_2^{C-j+1}\right) \frac{1}{C} = 1 - \frac{q_2}{C} \frac{q_1^C - q_2^C}{q_1 - q_2}, \quad (168)$$

which does not depend on the rebuild duration.

Corollary 12: It holds that

$$P_{\text{UF}|2} \approx \begin{cases} \dfrac{(C+1)(m-2)}{2} P_s, & \text{for } P_s \ll P_{s*}^{(2)} \\ 1, & \text{for } P_s \gg P_{s*}^{(2)}, \end{cases} \quad (169)$$

where $P_{s*}^{(2)}$ is determined by (162).

Proof: See Appendix E. ∎

Remark 26: It follows from (155) and (160) that $P_{\text{UF in } S_1|2} \ll P_{\text{UF in } S_2|2}$ because the former is of the order $O(P_s^2)$, whereas the latter is of the order $O(P_s)$. Consequently, in conjunction with two device failures, an unrecoverable failure is more likely to be encountered after the second device failure and not before. This in turn implies that

$$P_{\text{UF}|2} \approx P_{\text{UF in } S_2|2}, \quad (170)$$

which is confirmed by comparing (160) with (169).

A subsequent (third) device failure may occur during the rebuild process of the remaining $C-j$ codewords, the duration of which is equal to $[(C-j)/C]R$. The probability $P_{\text{DF}_3}(j|R)$ that one of the $m-2$ remaining devices in the array fails during this rebuild depends on its duration and the aggregate failure rate of the $m-2$ highly reliable devices, and is

$$P_{\text{DF}_3}(j|R) \approx \frac{C-j}{C}(m-2)\lambda R. \quad (171)$$

Consequently, for a rebuild time duration of R, the probability $P_{\text{DF},2}(j|R)$ of encountering a second device failure during reconstruction of the jth codeword and not encountering a third device failure during the remaining rebuild time is determined by the product of $P_{\text{DF}_2}(j|R)$ and $1 - P_{\text{DF}_3}(j|R)$, that is,

$$P_{\text{DF},2}(j|R) \approx \frac{(m-1)\lambda R}{C}\left[1 - \frac{C-j}{C}(m-2)\lambda R\right],$$
$$\text{for } j = 1, 2, \ldots, C. \quad (172)$$

Therefore, the probability $P_{\text{DF},2}(j)$ of encountering a second device failure during reconstruction of the jth codeword and not encountering a third device failure during the remaining rebuild time is obtained by unconditioning (172) on R using (4) as follows:

$$P_{\text{DF},2}(j) \approx \frac{(m-1)}{C}\frac{\lambda}{\mu}\left[1 - \frac{C-j}{C}(m-2)\frac{\lambda}{\mu}f_R\right],$$
$$\text{for } j = 1, 2, \ldots, C, \quad (173)$$

where

$$f_R \triangleq \frac{E(R^2)}{[E(R)]^2} \geq 1. \quad (174)$$

Thus, the probability $P_{\text{DF},2}$ of two device failures during rebuild is obtained by (173) as follows:

$$P_{\text{DF},2} \approx \sum_{j=1}^{C} P_{\text{DF},2}(j) \quad (175)$$

$$\approx (m-1)\frac{\lambda}{\mu}\left[1 - \frac{C-1}{C}(m-2)\frac{\lambda}{\mu}f_R\right] \quad (176)$$

$$\overset{(5)}{\approx} (m-1)\frac{\lambda}{\mu} \overset{(33)}{\approx} P_{\text{DF}_2}. \quad (177)$$

The probability $P_{\text{UF},2}$ of data loss due to unrecoverable failures in conjunction with two device failures is obtained by unconditioning (154) on j and using (173) as follows:

$$
\begin{aligned}
P_{\text{UF},2} &= \sum_{j=1}^{C} P_{\text{UF}|2}(j)\, P_{\text{DF},2}(j) \\
&\approx \sum_{j=1}^{C} (1 - q_1^{j-1}\, q_2^{C-j+1})\, \frac{(m-1)}{C}\, \frac{\lambda}{\mu} \\
&\qquad \cdot \left[1 - \frac{C-j}{C}\, (m-2)\, \frac{\lambda}{\mu}\, f_R \right] \\
&= \left(1 - \frac{q_2}{C}\, \frac{q_1^C - q_2^C}{q_1 - q_2} \right) (m-1)\, \frac{\lambda}{\mu} \\
&\quad - \left[\frac{C(C-1)}{2} - q_2^2\, \frac{q_1^C - C\, q_1\, q_2^{C-1} + (C-1)\, q_2^C}{(q_1 - q_2)^2} \right] \cdot \\
&\qquad \cdot \frac{(m-1)(m-2)}{C^2} \left(\frac{\lambda}{\mu} \right)^2 f_R\,,
\end{aligned}
$$

(178)

where f_R is determined by (174).

Corollary 13: It holds that

$$
P_{\text{UF},2} \approx
\begin{cases}
A\, P_s\,, & \text{for } P_s \ll P_{s*}^{(2)} \\
P_{\text{DF},2}\,, & \text{for } P_s \gg P_{s*}^{(2)}\,,
\end{cases}
$$

(179)

where

$$
A = \frac{C+1}{2} \left[1 - \frac{2}{3}\, \frac{C-1}{C}\, (m-2)\, \frac{\lambda}{\mu}\, f_R \right] (m-1)(m-2)\, \frac{\lambda}{\mu}
$$

(180)

$$
\approx \frac{C}{2} \left[1 - \frac{2}{3}\, (m-2)\, \frac{\lambda}{\mu}\, f_R \right] (m-1)(m-2)\, \frac{\lambda}{\mu}
$$

(181)

$$
\overset{(5)}{\approx} \frac{C}{2}\, (m-1)(m-2)\, \frac{\lambda}{\mu}\,,
$$

(182)

and $P_{\text{DF},2}$ and $P_{s*}^{(2)}$ are given by (176) and (162), respectively.

Proof: See Appendix E. ∎

Remark 27: Note that the minuend of the difference shown on the right-hand side of (178) is much greater than the subtrahend because the former is of the order $O(\lambda/\mu)$, whereas the latter is of the order $O((\lambda/\mu)^2)$ and is also further reduced owing to division by the large number C^2. Consequently, from (178), it follows that

$$
\begin{aligned}
P_{\text{UF},2} &\approx \left(1 - \frac{q_2}{C}\, \frac{q_1^C - q_2^C}{q_1 - q_2} \right) (m-1)\, \frac{\lambda}{\mu} \\
&\overset{(33)(168)}{\approx} P_{\text{UF}|2}\, P_{\text{DF}_2} \\
&\overset{(170)}{\approx} P_{\text{UF in } S_2|2}\, P_{\text{DF}_2} \\
&\overset{(33)(163)}{\approx} \left[1 + \frac{1 - q_2^C}{\log(q_2^C)} \right] (m-1)\, \frac{\lambda}{\mu}\,,
\end{aligned}
$$

(183)

(184)

(185)

(186)

where q_1 and q_2 are given by (126) and (147), respectively.

Remark 28: According to (131) and (162) and for large values of C, it holds that $P_{s*}^{(2)} \ll P_{s*}^{(4)}$. Consequently, from (142), (129), (179), and (182), and given that $P_{s*}^{(2)} \sim 1/C \ll \lambda/\mu$, it holds that

$$
P_{\text{UF},1} \ll P_{\text{UF},2}\,, \qquad \text{for } P_s \ll P_{s*}^{(2)}\,.
$$

(187)

We now proceed to assess the amount of data loss. As discussed above, each of the $C - j + 1$ codewords in $S_{2,j}$ loses an expected number of $E(L_2)$ symbols. Thus, the expected total number $E(S_{\text{U},2}^+ \,|\, \text{DF}_2 \text{ at } j)$ of symbols stored in these $C - j + 1$ codewords and lost due to unrecoverable failures is

$$
E(S_{\text{U},2}^+ \,|\, \text{DF}_2 \text{ at } j) = (C - j + 1)\, E(L_2)\,,
$$

(188)

where $E(L_2)$ is determined by (149). Furthermore, each of the $j - 1$ codewords in $S_{1,j}$ loses an expected number of $E(L_1)$ symbols. Consequently, the expected total number $E(S_{\text{U},2}^- \,|\, \text{DF}_2 \text{ at } j)$ of symbols stored in these $j-1$ codewords and lost due to unrecoverable failures is

$$
E(S_{\text{U},2}^- \,|\, \text{DF}_2 \text{ at } j) = (j - 1)\, E(L_1)\,,
$$

(189)

where $E(L_1)$ is determined by (134). Unconditioning (188) and (189) on the event of a device failure during the reconstruction of the jth codeword, and using (173), yields after some manipulations

$$
\begin{aligned}
E(S_{\text{U},2}^+) &= \sum_{j=1}^{C} E(S_{\text{U},2}^+ \,|\, \text{DF}_2 \text{ at } j) P_{\text{DF},2}(j) \\
&= \sum_{j=1}^{C} (C - j + 1)\, E(L_2)\, P_{\text{DF},2}(j) \\
&\approx \frac{1}{2}\, (C + 1)\, E(L_2)\, (m-1)\, \frac{\lambda}{\mu} \\
&\quad - \frac{(C+1)\, E(L_2)}{3} \cdot \frac{C-1}{C}\, (m-1)\, (m-2) \left(\frac{\lambda}{\mu} \right)^2 f_R\,,
\end{aligned}
$$

(190)

and

$$
\begin{aligned}
E(S_{\text{U},2}^-) &= \sum_{j=1}^{C} E(S_{\text{U},2}^- \,|\, \text{DF}_2 \text{ at } j) P_{\text{DF},2}(j) \\
&= \sum_{j=1}^{C} (j - 1)\, E(L_1)\, P_{\text{DF},2}(j) \\
&\approx \frac{1}{2}\, (C - 1)\, E(L_1)\, (m-1)\, \frac{\lambda}{\mu} \\
&\quad - \frac{(C-2)\, E(L_1)}{6} \cdot \frac{C-1}{C}\, (m-1)\, (m-2) \left(\frac{\lambda}{\mu} \right)^2 f_R\,,
\end{aligned}
$$

(191)

where $E(L_1)$, $E(L_2)$, and f_R are determined by (134), (149), and (174), respectively.

The expected total number $E(S_{\text{U},2})$ of symbols lost due to unrecoverable errors in conjunction with two device failures is

$$
E(S_{\text{U},2}) = E(S_{\text{U},2}^+) + E(S_{\text{U},2}^-)\,,
$$

(192)

where $E(S_{\text{U},2}^+)$ and $E(S_{\text{U},2}^-)$ are determined by (190) and (191), respectively.

Remark 29: From (135) and (150), it follows that $E(L_1) \ll E(L_2)$ because the former is of the order $O(P_s^2)$, whereas the latter is of the order $O(P_s)$. From (190) and (191), it follows that $E(S_{\text{U},2}^+) \sim E(L_2)$ and $E(S_{\text{U},2}^-) \sim E(L_1)$, respectively. Thus,

$$
E(S_{\text{U},2}^+) \gg E(S_{\text{U},2}^-)\,.
$$

(193)

Corollary 14: It holds that

$$E(S_{\mathrm{U},2}) \approx E(S_{\mathrm{U},2}^{+}) \,. \tag{194}$$

Proof: Immediate from (192) and (193). ∎

Remark 30: For small values of P_s and large values of C, from (190) and (194), and using (5) and (150), it follows that

$$E(S_{\mathrm{U},2}) \approx \frac{1}{2} \, C \, E(L_2) \, (m-1) \, \frac{\lambda}{\mu} \tag{195}$$

$$\approx \frac{3}{2} \, C \, (m-1)(m-2) \, \frac{\lambda}{\mu} \, P_s \,, \quad \text{for } P_s \ll \frac{1}{m-2} \,. \tag{196}$$

Remark 31: From (143) and (191), it follows that $E(S_{\mathrm{U},1}) \gg E(S_{\mathrm{U},2})$ because $E(S_{\mathrm{U},2})$ is of the order $O(\lambda/\mu)$, which is very small, whereas $E(S_{\mathrm{U},1})$ is not. Furthermore, from (144) and (196), it follows that

$$E(S_{\mathrm{U},1}) \ll E(S_{\mathrm{U},2})$$
$$\Leftrightarrow \frac{3}{2} \, C \, (m-1) \, (m-2) \, P_s^2 \ll \frac{3}{2} \, C \, (m-1) \, (m-2) \, \frac{\lambda}{\mu} \, P_s$$
$$\Leftrightarrow P_s \ll P_{s*}^{(5)} \,, \tag{197}$$

where

$$\boxed{P_{s*}^{(5)} \triangleq \frac{\lambda}{\mu}} \,. \tag{198}$$

Consequently, for very small values of P_s, the symbols lost due to unrecoverable failures are predominately encountered during the rebuild phase after a second device failure.

C. Three Device Failures

As discussed in Section VI-B, after a second device failure during reconstruction of the jth ($1 \le j \le C$) codeword, a subsequent (third) device failure may occur during the rebuild process of the remaining $C - j$ codewords, which form the $S_{2,j}$ set. In particular, such a failure may occur during reconstruction of the ith ($j+1 \le i \le C$) codeword, which divides the codewords of the $S_{2,j}$ set into two subsets:

$S_{2,j,i}^{-}$: the set of $i-j$ codewords in $S_{2,j}$ already considered for reconstruction prior to the third device failure, and

$S_{2,j,i}^{+}$: the set of the remaining $C - i + 1$ codewords in $S_{2,j}$, none of which can be reconstructed.

Analogous to (32), the probability $P_{\mathrm{DF}_3}(i|R)$ that a third device failure occurs during reconstruction of the ith ($j+1 \le i \le C$) codeword during the rebuild process in conjunction with two device failures, and given a rebuild time of R, is

$$P_{\mathrm{DF}_3}(i|j,R) \approx \frac{(m-2) \, \lambda \, R}{C} \,, \quad \text{for } i = j+1, j+2, \ldots, C \,. \tag{199}$$

Consequently, the probability $P_{j,i}(R)$ of a second device failure during reconstruction of the jth codeword and a third device failure during reconstruction of the ith codeword, and given a rebuild time of R, is the product of $P_{\mathrm{DF}_2}(j|R)$ and $P_{\mathrm{DF}_3}(i|j,R)$ obtained from (32) and (199) as follows:

$$P_{j,i}(R) \triangleq P_{\mathrm{DF}_2}(j|R) \, P_{\mathrm{DF}_3}(i|j,R) \approx \frac{(m-1)(m-2)}{C^2} \, \lambda^2 \, R^2 \,,$$
$$\text{for } j = 1, 2, \ldots, C \text{ and } i = j+1, j+2, \ldots, C \,. \tag{200}$$

Therefore, the probability $P_{j,i}$ of a second device failure during reconstruction of the jth codeword and a third device failure during reconstruction of the ith codeword is obtained by unconditioning (200) on R and using (4) and (174) as follows:

$$P_{j,i} = E(P_{j,i}(R)) \approx \frac{(m-1)(m-2)}{C^2} \left(\frac{\lambda}{\mu}\right)^2 f_R \,,$$
$$\text{for } j = 1, 2, \ldots, C \text{ and } i = j+1, j+2, \ldots, C \,. \tag{201}$$

Also, the probability $P_{\mathrm{DF}_3}(j|R)$ that, during reconstruction of the codewords in $S_{2,j}$, one of the $m - 2$ remaining devices in the array fails is obtained from (199) as follows:

$$P_{\mathrm{DF}_3}(j|R) = \sum_{i=j+1}^{C} P_{\mathrm{DF}_3}(i|j,R) \approx \sum_{i=j+1}^{C} \frac{(m-2) \, \lambda \, R}{C}$$
$$= (C-j) \, \frac{(m-2) \, \lambda \, R}{C} \,, \tag{202}$$

which is in agreement with (171).

The probability $P_{\mathrm{DF}_3}(R)$ of encountering three device failures given a rebuild duration of R is obtained by unconditioning (202) on j and using (32) as follows:

$$P_{\mathrm{DF}_3}(R) \approx \sum_{j=1}^{C} P_{\mathrm{DF}_3}(j|R) \, P_{\mathrm{DF}_2}(j|R)$$
$$\approx \sum_{j=1}^{C} \frac{C-j}{C^2} \, (m-1)(m-2) \, \lambda^2 \, R^2$$
$$\approx \frac{C-1}{2\,C} \, (m-1)(m-2) \, \lambda^2 \, R^2 \,. \tag{203}$$

The probability P_{DF_3} of data loss due to three device failures, that is, the probability of two device failures during the rebuild process is obtained by unconditioning (203) on R and using (4) and (174) as follows:

$$P_{\mathrm{DF}_3} \approx \frac{C-1}{2\,C} \, (m-1)(m-2) \left(\frac{\lambda}{\mu}\right)^2 f_R \,. \tag{204}$$

In particular, for large values of C, it holds that

$$P_{\mathrm{DF}_3} \approx \frac{(m-1)(m-2)}{2} \left(\frac{\lambda}{\mu}\right)^2 f_R \,, \tag{205}$$

Note that the occurrence of the third device failure does not exclude the possibility of unrecoverable failures being encountered prior to its occurrence. Also, each of the $C - i + 1$ codewords in $S_{2,j,i}^{+}$ can no longer be reconstructed. In particular, the three symbols of the ith codeword stored on the three failed devices can no longer be recovered and are lost. Furthermore, any of its remaining $m - 3$ symbols with unrecoverable errors will also be lost, leading to unrecoverable failure. Thus, the probability p_3 of not encountering an unrecoverable failure due to unrecoverable errors in the remaining $m - 3$ symbols is

$$p_3 = (1 - P_s)^{m-3} \,, \tag{206}$$

which for very small values of P_s implies that

$$p_3 \approx \begin{cases} 1 - (m-3) \, P_s \,, & \text{for } P_s \ll \frac{1}{m-3} \\ 0 \,, & \text{for } P_s \gg \frac{1}{m-3} \,. \end{cases} \tag{207}$$

The same applies for the remaining $C - i$ codewords. Therefore, the probability that no unrecoverable failures occur in $S_{1,j}$ is q_1^{j-1}, the probability that no unrecoverable failures occur in $S_{2,j,i}^-$ is q_2^{i-j}, and the probability that no unrecoverable failures occur in $S_{2,j,i}^+$ is p_3^{C-i+1}. Consequently, the probability $P_{\mathrm{UF}|3}(j,i)$ that an unrecoverable failure occurs, given three device failures, is

$$P_{\mathrm{UF}|3}(j,i) = 1 - q_1^{j-1}\, q_2^{i-j}\, p_3^{C-i+1} \; . \tag{208}$$

The probability $P_{\mathrm{UF},3}$ of data loss due to unrecoverable failures in conjunction with three device failures is determined by the following proposition.

Proposition 1: It holds that

$$P_{\mathrm{UF},3} = \left[\frac{C\,(C-1)}{2} - \frac{q_2\,p_3}{p_3 - q_2} \left(\frac{q_1^C - p_3^C}{q_1 - p_3} - \frac{q_1^C - q_2^C}{q_1 - q_2} \right) \right] \cdot$$
$$\cdot \frac{(m-1)(m-2)}{C^2} \left(\frac{\lambda}{\mu} \right)^2 f_R \; . \tag{209}$$

Proof: See Appendix F. ∎

Corollary 15: It holds that

$$P_{\mathrm{UF},3} \approx \begin{cases} B\,P_s \; , & \text{for } P_s \ll P_{s*}^{(2)} \\ P_{\mathrm{DF}_3} \; , & \text{for } P_s \gg P_{s*}^{(2)} \; , \end{cases} \tag{210}$$

where

$$B = \frac{(C-1)(C+1)}{6\,C}\,(2m-5)(m-1)(m-2)\left(\frac{\lambda}{\mu}\right)^2 f_R \tag{211}$$

$$\approx \frac{C}{6}\,(2m-5)(m-1)(m-2)\left(\frac{\lambda}{\mu}\right)^2 f_R \; , \tag{212}$$

and P_{DF_3} and $P_{s*}^{(2)}$ are given by (205) and (162), respectively.

Proof: See Appendix G. ∎

Remark 32: From (210), it follows that for $P_s \gg P_{s*}^{(2)}$, $P_{\mathrm{UF}|3} = P_{\mathrm{UF},3}/P_{\mathrm{DF}_3} \approx 1$, which implies that a third device failure leads to data loss owing to unrecoverable failures. In this case, the probability of data loss is equal to that of a RAID-6 system in the absence of latent errors, that is, $P_{\mathrm{UF},3} = P_{\mathrm{DF}_3}$.

Remark 33: From (179) and (210), it follows that

$$P_{\mathrm{UF},2} \gg P_{\mathrm{UF},3} \; . \tag{213}$$

According to (182) and (212), it holds that $A \gg B$ because the former is of the order $O(\lambda/\mu)$, whereas the latter is of the order $O((\lambda/\mu)^2)$, and similarly, according to (177) and (205), $P_{\mathrm{DF}_2} \approx P_{\mathrm{DF}_2} \gg P_{\mathrm{DF}_3}$.

We now proceed to assess the amount of data loss. As discussed above, the three symbols of each of the $C - i + 1$ codewords in $S_{2,j,i}^+$ stored on the three failed devices can no longer be recovered and are lost. Thus, the total number $S_{\mathrm{D}}(j,i)$ of symbols stored on the three failed devices and lost is

$$S_{\mathrm{D}}(j,i) = 3\,(C+1-i) \; . \tag{214}$$

Also, the expected number $E(S_{\mathrm{U},3}^+ \,|\, \mathrm{DF}_2 \text{ at } j \text{ and } \mathrm{DF}_3 \text{ at } i)$ of symbols stored in these $C - i + 1$ codewords and lost owing to unrecoverable errors is

$$E(S_{\mathrm{U},3}^+ \,|\, \mathrm{DF}_2 \text{ at } j \text{ and } \mathrm{DF}_3 \text{ at } i) = (C+1-i)\,(m-3)\,P_s \; . \tag{215}$$

Moreover, each of the $i - j$ codewords in $S_{2,j,i}^-$ loses an expected number of $E(L_2)$ symbols. Consequently, the expected total number $E(S_{\mathrm{U},3}^- \,|\, \mathrm{DF}_2 \text{ at } j \text{ and } \mathrm{DF}_3 \text{ at } i)$ of symbols stored in these $j - i$ codewords and lost due to unrecoverable errors is

$$E(S_{\mathrm{U},3}^- \,|\, \mathrm{DF}_2 \text{ at } j \text{ and } \mathrm{DF}_3 \text{ at } i) = (i-j)\,E(L_2) \; . \tag{216}$$

Furthermore, each of the $j - 1$ codewords in $S_{1,j}$ loses an expected number of $E(L_1)$ symbols. Consequently, the expected total number $E(S_{\mathrm{U},3}^\odot \,|\, \mathrm{DF}_2 \text{ at } j \text{ and } \mathrm{DF}_3 \text{ at } i)$ of symbols stored in these $j - 1$ codewords and lost due to unrecoverable errors is

$$E(S_{\mathrm{U},3}^\odot \,|\, \mathrm{DF}_2 \text{ at } j \text{ and } \mathrm{DF}_3 \text{ at } i) = (j-1)\,E(L_1) \; . \tag{217}$$

Unconditioning (214), (215), (216), and (217) on the event of two device failures during the reconstruction of the jth and ith codewords, and using (201), yields

$$E(S_{\mathrm{D}}) = \sum_{j=1}^{C} \sum_{i=j+1}^{C} S_{\mathrm{D}}(j,i)\,P_{j,i}$$
$$\approx \frac{(C-1)\,(C+1)}{2\,C}\,(m-1)\,(m-2)\left(\frac{\lambda}{\mu}\right)^2 f_R \; , \tag{218}$$

$$E(S_{\mathrm{U},3}^+) = \sum_{j=1}^{C} \sum_{i=j+1}^{C} E(S_{\mathrm{U},3}^+ \,|\, \mathrm{DF}_2 \text{ at } j \text{ and } \mathrm{DF}_3 \text{ at } i)\,P_{j,i}$$
$$\approx \frac{(C-1)\,(C+1)}{6\,C}\,(m-1)\,(m-2)\,(m-3)\left(\frac{\lambda}{\mu}\right)^2 f_R\,P_s \; , \tag{219}$$

$$E(S_{\mathrm{U},3}^-) = \sum_{j=1}^{C} \sum_{i=j+1}^{C} E(S_{\mathrm{U},3}^- \,|\, \mathrm{DF}_2 \text{ at } j \text{ and } \mathrm{DF}_3 \text{ at } i)\,P_{j,i}$$
$$\approx \frac{(C-1)\,(C+1)}{6\,C}\,(m-1)\,(m-2)\left(\frac{\lambda}{\mu}\right)^2 f_R\,E(L_2) \; , \tag{220}$$

and

$$E(S_{\mathrm{U},3}^\odot) = \sum_{j=1}^{C} \sum_{i=j+1}^{C} E(S_{\mathrm{U},3}^\odot \,|\, \mathrm{DF}_2 \text{ at } j \text{ and } \mathrm{DF}_3 \text{ at } i)\,P_{j,i}$$
$$\approx \frac{(C-1)\,(C-2)}{6\,C}\,(m-1)\,(m-2)\left(\frac{\lambda}{\mu}\right)^2 f_R\,E(L_1) \; , \tag{221}$$

where $E(L_1)$, $E(L_2)$, and f_R are determined by (134), (149), and (174), respectively.

In particular, from (218) and considering that, for large values of C, $C + 1 \approx C$, it holds that

$$E(S_{\mathrm{D}}) \approx \frac{C}{2}\,(m-1)\,(m-2)\left(\frac{\lambda}{\mu}\right)^2 f_R \; . \tag{222}$$

The expected total number $E(S_{U,3})$ of symbols lost due to unrecoverable errors in conjunction with two device failures is

$$E(S_{U,3}) = E(S_{U,3}^+) + E(S_{U,3}^-) + E(S_{U,3}^\odot) , \qquad (223)$$

where $E(S_{U,3}^+)$, $E(S_{U,3}^-)$, and $E(S_{U,3}^\odot)$ are determined by (219), (220), and (221), respectively.

Remark 34: From (219), (220), and (221), and by virtue of (135) and (150), it follows that $E(S_{U,3}^-) > E(S_{U,3}^+)$ and $E(S_{U,3}^-) > E(S_{U,3}^\odot)$. Also, from (195) and (220), it follows that $E(S_{U,2}) \gg E(S_{U,3}^-)$ because $E(S_{U,2})$ is of the order $O(\lambda/\mu)$, whereas $E(S_{U,3}^-)$ is of the order $O((\lambda/\mu)^2)$. From the discussion above, it follows that $E(S_{U,2}) \gg E(S_{U,3})$.

From (140), (143), (196), and (222), it follows that

$$E(S_{U,1}) \ll E(S_D) \iff P_s \ll \sqrt{\frac{f_R}{3}} P_{s*}^{(5)} , \qquad (224)$$

$$E(S_{U,2}) \ll E(S_D) \iff P_s \ll \frac{f_R}{3} P_{s*}^{(5)} , \qquad (225)$$

where $P_{s*}^{(5)}$ is determined by (198).

Remark 35: From (81) and (198), it follows that $P_s^{(3)}$ and $P_{s*}^{(5)}$ are of the same order.

Remark 36: From (218), it follows that the expected number $E(S_D)$ of symbols stored on the three failed devices and lost is of the order $O((\lambda/\mu)^2)$. Note that the above analysis does not exclude the possibility that additional device failures occur during rebuild. However, the corresponding expected number of additional lost symbols can be ignored because it is of the order $O((\lambda/\mu)^3)$, which is much smaller than $O((\lambda/\mu)^2)$.

D. Data Loss

Data loss during rebuild may occur because of an unrecoverable failure of one or more codewords in case of one, two or three device failures. These three mutually exclusive events imply that

$$P_{DL} = P_{UF,1} + P_{UF,2} + P_{DF_3} . \qquad (226)$$

Substituting (137), (178), and (204) into (226) yields

$$P_{DL} \approx \left[1 - (m-1)\frac{\lambda}{\mu} \right] (1 - q_1^C)$$
$$+ \left(1 - \frac{q_2}{C} \frac{q_1^C - q_2^C}{q_1 - q_2} \right) (m-1)\frac{\lambda}{\mu}$$
$$- \left[\frac{(C-1)C}{2} - q_2^2 \frac{q_1^C - C q_1 q_2^{C-1} + (C-1) q_2^C}{(q_1 - q_2)^2} \right] .$$
$$\cdot \frac{(m-1)(m-2)}{C^2} \left(\frac{\lambda}{\mu} \right)^2 f_R ,$$
$$+ \frac{C-1}{2C} (m-1)(m-2) \left(\frac{\lambda}{\mu} \right)^2 f_R . \qquad (227)$$

Thus, after some manipulations, (227) yields

$$P_{DL} \approx 1 - q_1^C \left[1 - (m-1)\frac{\lambda}{\mu} \right] - \frac{q_2}{C} \frac{q_1^C - q_2^C}{q_1 - q_2} (m-1)\frac{\lambda}{\mu}$$
$$+ q_2^2 \frac{q_1^C - C q_1 q_2^{C-1} + (C-1)q_2^C}{(q_1 - q_2)^2} \frac{(m-1)(m-2)}{C^2} \left(\frac{\lambda}{\mu} \right)^2 f_R ,$$
$$\qquad (228)$$

where q_1 and q_2 are given by (126) and (147), respectively.

Substituting (126) and (147) into (228) yields

$$P_{DL} \approx 1 - (1 - P_s)^{C(m-2)}$$
$$\cdot \left\{ [1 + (m-2)P_s]^C \right.$$
$$+ \frac{m-1}{C(m-2)P_s} \frac{\lambda}{\mu} \left[[1 + (m-2)P_s]^C [1 - C(m-2)P_s] \right.$$
$$\left. \left. - \frac{[1 + (m-2)P_s]^C - C(m-2)P_s - 1}{C P_s} \frac{\lambda}{\mu} f_R \right] \right\} .$$
$$\qquad (229)$$

Corollary 16: For $P_s \ll P_{s*}^{(2)}$, it holds that

$$P_{DL} \approx P_{DF_3} + P_{UF,2} \overset{(179)}{\approx} P_{DF_3} + A P_s \qquad (230)$$
$$\overset{(205)(182)}{\approx} \frac{(m-1)(m-2)}{2} \frac{\lambda}{\mu} \left(\frac{\lambda}{\mu} f_R + C P_s \right), \quad (231)$$

where P_{DF_3} and A are given by (204) and (180), respectively.

Proof: Immediate from (226) by considering (187). ∎

Remark 37: It follows from (179), (182), and (205) that the range $[0, P_{s*}^{(1)})$ of P_s in which the probability $P_{UF,2}$ is much smaller than the probability P_{DF_3} of two device failures occurring during rebuild is obtained by

$$P_{UF,2} \ll P_{DF_3} \iff A P_s \ll \frac{(m-1)(m-2)}{2} \left(\frac{\lambda}{\mu} \right)^2 f_R$$
$$\iff P_s \ll P_{s*}^{(1)} , \qquad (232)$$

where

$$\boxed{ P_{s*}^{(1)} \triangleq \frac{1}{C} \cdot \frac{\lambda}{\mu} \cdot f_R } . \qquad (233)$$

From the above, it follows that $P_{UF,2} \gg P_{DF_3}$ for $P_s \gg P_{s*}^{(1)}$ and, therefore, P_{DL} is dominated by $P_{UF,2}$.

Remark 38: From (169), it follows that $P_{UF|2} \approx 1$ for $P_s \gg P_{s*}^{(2)}$, which implies that a second device failure leads to data loss owing to unrecoverable failures. In this case, the probability of data loss is equal to that of a RAID-5 system in the absence of latent errors, that is, $P_{UF,2} = P_{DF_2}$, as derived in (179) and (177).

Remark 39: From (131) and (162), it follows that $P_{s*}^{(2)} \ll P_{s*}^{(4)}$. Consequently, the range $[0, P_{s*}^{(3)})$ of P_s in which probability $P_{UF,2}$ is much greater than probability $P_{UF,1}$ is obtained from (129), (142), and (179) as follows:

$$P_{UF,1} \ll P_{UF,2}$$
$$\iff \begin{cases} C \frac{(m-1)(m-2)}{2} P_s^2 \ll A P_s , & \text{for } P_s \ll P_{s*}^{(2)} \ll P_{s*}^{(4)} \\ C \frac{(m-1)(m-2)}{2} P_s^2 \ll P_{DF,2} , & \text{for } P_{s*}^{(2)} \ll P_s \ll P_{s*}^{(4)} \end{cases}$$
$$\qquad (234)$$
$$\overset{(177)(182)}{\iff} \begin{cases} C \frac{(m-1)(m-2)}{2} P_s^2 \ll \frac{C}{2}(m-1)(m-2)\frac{\lambda}{\mu} P_s \\ C \frac{(m-1)(m-2)}{2} P_s^2 \ll (m-1)\frac{\lambda}{\mu} \end{cases}$$
$$\qquad (235)$$
$$\iff P_s \ll P_{s*}^{(3)} , \qquad (236)$$

where, by virtue of (162),

$$P_{s^*}^{(3)} \triangleq \begin{cases} \dfrac{\lambda}{\mu}, & \text{for } \dfrac{\lambda}{\mu} \leq \dfrac{1}{C} \cdot \dfrac{2}{m-2} = P_{s^*}^{(2)} \\ \sqrt{\dfrac{2}{C(m-2)} \dfrac{\lambda}{\mu}}, & \text{for } \dfrac{\lambda}{\mu} \geq \dfrac{1}{C} \cdot \dfrac{2}{m-2} = P_{s^*}^{(2)} \end{cases}.$$ (237)

From the above discussion, it follows that for $P_s \gg P_{s^*}^{(3)}$, P_{DL} is dominated by $P_{\mathrm{UF},1}$.

Corollary 17: For $P_s \gg P_{s^*}^{(2)}$, it holds that

$$P_{\mathrm{DL}} \approx P_{\mathrm{UF},1} + P_{\mathrm{UF},2}$$ (238)

$$\overset{(138)(179)}{\approx} \begin{cases} P_{\mathrm{DF},2} + P_{\mathrm{DF},1} \dfrac{(m-1)(m-2)}{2} C P_s^2, & \text{for } P_s \ll P_{s^*}^{(4)} \\ P_{\mathrm{DF},2} + P_{\mathrm{DF},1}, & \text{for } P_s \gg P_{s^*}^{(4)} \end{cases}$$ (239)

$$\overset{(34)(177)}{\approx} \begin{cases} (m-1)\left[\dfrac{\lambda}{\mu} + \dfrac{C(m-2)}{2}P_s^2\right], & \text{for } P_s \ll P_{s^*}^{(4)} \\ 1, & \text{for } P_s \gg P_{s^*}^{(4)} \end{cases}$$ (240)

where $P_{\mathrm{DF},1}$ and $P_{\mathrm{DF},2}$ are determined by (34) and (176), respectively.

Proof: Immediate from (226) by considering (232) and $P_s \gg P_{s^*}^{(2)} \gg P_{s^*}^{(1)}$ and $P_{\mathrm{DF},1} \approx 1$. ∎

Corollary 18: For small values of λ/μ, it holds that

$$P_{\mathrm{DL}} \approx P_{\mathrm{UF}|1} + P_{\mathrm{UF},2} + P_{\mathrm{DF}_3}$$ (241)

$$\overset{(184)}{\approx} P_{\mathrm{UF}|1} + P_{\mathrm{UF}|2} P_{\mathrm{DF}_2} + P_{\mathrm{DF}_3}$$ (242)

$$\overset{(128)(186)(205)}{\approx} 1 - q_1^C + \left[1 + \dfrac{1 - q_2^C}{\log(q_2^C)}\right](m-1)\dfrac{\lambda}{\mu}$$
$$+ \dfrac{(m-1)(m-2)}{2}\left(\dfrac{\lambda}{\mu}\right)^2 f_R.$$ (243)

where q_1, q_2, and f_R are determined by (126), (147), and (174) respectively.

Proof: Immediate by substituting (142) into (226). ∎

According to (162) and (237), it holds that $P_{s^*}^{(2)} \lessgtr P_{s^*}^{(3)}$ $\Leftrightarrow \lambda/\mu \gtrless 2/[C(m-2)] = P_{s^*}^{(2)}$. Depending on the values of λ/μ, m and C, we consider the following two cases:

Case 1: $\frac{\lambda}{\mu} \geq \frac{1}{C} \cdot \frac{2}{m-2} = P_{s^*}^{(2)}$. From (233), (162), (237), and (131), it holds that $P_{s^*}^{(1)} \ll P_{s^*}^{(2)} \leq P_{s^*}^{(3)} < P_{s^*}^{(4)}$. Also, from (241), and considering (129), (177), (179), (182), and (205), it follows that

$$P_{\mathrm{DL}} \approx \begin{cases} \dfrac{(m-1)(m-2)}{2}\left(\dfrac{\lambda}{\mu}\right)^2 f_R, & \text{for } P_s \ll P_{s^*}^{(1)} \\ C\dfrac{(m-1)(m-2)}{2}\dfrac{\lambda}{\mu}P_s, & \text{for } P_{s^*}^{(1)} \ll P_s \ll P_{s^*}^{(2)} \\ (m-1)\dfrac{\lambda}{\mu}, & \text{for } P_{s^*}^{(2)} \ll P_s \ll P_{s^*}^{(3)} \\ C\dfrac{(m-1)(m-2)}{2}P_s^2, & \text{for } P_{s^*}^{(3)} \ll P_s \ll P_{s^*}^{(4)} \\ 1, & \text{for } P_s \gg P_{s^*}^{(4)} \end{cases}$$ (244)

where $P_{s^*}^{(1)}$, $P_{s^*}^{(2)}$, $P_{s^*}^{(3)}$, and $P_{s^*}^{(4)}$ are determined by (233), (162), (237), and (131), respectively.

Case 2: $\frac{\lambda}{\mu} < \frac{1}{C} \cdot \frac{2}{m-2} = P_{s^*}^{(2)}$. From (233), (162), (237), (131), and (198), it holds that $P_{s^*}^{(1)} \ll P_{s^*}^{(3)} = P_{s^*}^{(5)} = \lambda/\mu < P_{s^*}^{(2)} < P_{s^*}^{(4)}$. Also, from (241), and considering (129), (179), (182), and (205), it follows that

$$P_{\mathrm{DL}} \approx \begin{cases} \dfrac{(m-1)(m-2)}{2}\left(\dfrac{\lambda}{\mu}\right)^2 f_R, & \text{for } P_s \ll P_{s^*}^{(1)} \\ C\dfrac{(m-1)(m-2)}{2}\dfrac{\lambda}{\mu}P_s, & \text{for } P_{s^*}^{(1)} \ll P_s \ll P_{s^*}^{(5)} \\ C\dfrac{(m-1)(m-2)}{2}P_s^2, & \text{for } P_{s^*}^{(5)} \ll P_s \ll P_{s^*}^{(4)} \\ 1, & \text{for } P_s \gg P_{s^*}^{(4)} \end{cases}$$ (245)

where $P_{s^*}^{(1)}$, $P_{s^*}^{(2)}$, $P_{s^*}^{(4)}$, and $P_{s^*}^{(5)}$ are determined by (233), (162), (131), and (198), respectively.

Remark 40: In the first case, that is, when $\lambda/\mu \geq 2/[C(m-2)]$, (244) implies that P_{DL}, as a function of P_s, exhibits three plateaus in the intervals $[0, P_{s^*}^{(1)})$, $(P_{s^*}^{(2)}, P_{s^*}^{(3)})$, and $(P_{s^*}^{(4)}, 1]$, respectively. However, in the second case, that is, when $\lambda/\mu < 2/[C(m-2)]$, it holds that $P_{s^*}^{(3)} < P_{s^*}^{(2)}$ and therefore the second plateau vanishes. In this case, (244) degenerates to (245).

Unrecoverable failures may occur in conjunction with one, two or three device failures. Consequently, the probability P_{UF} of one or more unrecoverable failures during rebuild is

$$P_{\mathrm{UF}} = P_{\mathrm{UF},1} + P_{\mathrm{UF},2} + P_{\mathrm{UF},3}$$ (246)

$$\overset{(213)}{\approx} P_{\mathrm{UF},1} + P_{\mathrm{UF},2}$$ (247)

$$\overset{(142)}{\approx} P_{\mathrm{UF}|1} + P_{\mathrm{UF},2}$$ (248)

$$\overset{(128)(186)}{\approx} 1 - q_1^C + \left[1 + \dfrac{1 - q_2^C}{\log(q_2^C)}\right](m-1)\dfrac{\lambda}{\mu},$$ (249)

where $P_{\mathrm{UF},1}$, $P_{\mathrm{UF},2}$, and $P_{\mathrm{UF},3}$ are obtained by (137), (178), and (209), respectively. Also, q_1 and q_2 are determined by (126) and (147), respectively. We proceed by considering the previous two cases:

Case 1: $\frac{\lambda}{\mu} \geq \frac{1}{C} \cdot \frac{2}{m-2} = P_{s^*}^{(2)}$. From (248) and considering (129), (177), (179), and (182), it follows that

$$P_{\mathrm{UF}} \approx \begin{cases} C\dfrac{(m-1)(m-2)}{2}\dfrac{\lambda}{\mu}P_s, & \text{for } P_s \ll P_{s^*}^{(2)} \\ (m-1)\dfrac{\lambda}{\mu}, & \text{for } P_{s^*}^{(2)} \ll P_s \ll P_{s^*}^{(3)} \\ C\dfrac{(m-1)(m-2)}{2}P_s^2, & \text{for } P_{s^*}^{(3)} \ll P_s \ll P_{s^*}^{(4)} \\ 1, & \text{for } P_s \gg P_{s^*}^{(4)} \end{cases}$$ (250)

where $P_{s^*}^{(1)}$, $P_{s^*}^{(2)}$, $P_{s^*}^{(3)}$, and $P_{s^*}^{(4)}$ are determined by (233), (162), (237), and (131), respectively.

Case 2: $\frac{\lambda}{\mu} < \frac{1}{C} \cdot \frac{2}{m-2} = P_{s^*}^{(2)}$. From (248) and considering (129), (179), and (182), it follows that

$$P_{\mathrm{UF}} \approx \begin{cases} C\dfrac{(m-1)(m-2)}{2}\dfrac{\lambda}{\mu}P_s, & \text{for } P_s \ll P_{s^*}^{(5)} \\ C\dfrac{(m-1)(m-2)}{2}P_s^2, & \text{for } P_{s^*}^{(5)} \ll P_s \ll P_{s^*}^{(4)} \\ 1, & \text{for } P_s \gg P_{s^*}^{(4)} \end{cases}$$ (251)

where $P_{s^*}^{(4)}$ and $P_{s^*}^{(5)}$ are given by (131) and (198), respectively.

E. Amount of Data Loss

As discussed in Section VI-D, data loss during rebuild may occur because of two additional (second and third) device failures or an unrecoverable failure of one or more codewords, or a combination thereof. Note that in all cases, data loss cannot involve only parity data, but also loss of user data.

Data loss during rebuild may occur because of unrecoverable failures in conjunction with one, two or three device failures. Consequently, the expected number $E(S_U)$ of symbols lost due to unrecoverable failures is obtained as follows:

$$E(S_U) = E(S_{U,1}) + E(S_{U,2}) + E(S_{U,3}) \,, \qquad (252)$$

where $E(S_{U,1})$, $E(S_{U,2})$, and $E(S_{U,3})$ are determined by (141), (192), and (223), respectively. Moreover, according to (197) and Remark 34, it holds that

$$E(S_U) \approx \begin{cases} E(S_{U,2})\,, & \text{for } P_s \ll P_{s^*}^{(5)} \\ E(S_{U,1})\,, & \text{for } P_s \gg P_{s^*}^{(5)} \end{cases} \qquad (253)$$

$$\overset{(144)(196)}{\approx} \begin{cases} \frac{3}{2}\,C\,(m-1)(m-2)\,\frac{\lambda}{\mu}\,P_s & \text{for } P_s \ll P_{s^*}^{(5)} \\ \frac{3}{2}\,C\,(m-1)\,(m-2)\,P_s^2\,, & \text{for } P_{s^*}^{(5)} \ll P_s \ll \frac{1}{m-1}, \end{cases}$$
$$\qquad (254)$$

where $P_{s^*}^{(5)}$ is determined by (198).

The expected total number $E(S)$ of symbols lost is

$$E(S) = E(S_D) + E(S_U) \,, \qquad (255)$$

where $E(S_D)$ and $E(S_U)$ are determined by (218) and (252), respectively.

Remark 41: It follows from (143), (195), (222), and (252) that

$$E(S) \approx C\left[E(L_1) + \frac{1}{2}\,E(L_2)\,(m-1)\,\frac{\lambda}{\mu} \right.$$
$$\left. + \frac{(m-1)(m-2)}{2}\,\left(\frac{\lambda}{\mu}\right)^2\,f_R \right] \qquad (256)$$

$$\overset{(135)(150)}{\approx} C\,\frac{(m-1)(m-2)}{2}\left[\left(\frac{\lambda}{\mu}\right)^2 f_R + 3\,\frac{\lambda}{\mu}\,P_s + 3\,P_s^2 \right],$$
$$\text{for } P_s \ll \frac{1}{m-1}\,, \qquad (257)$$

where $E(L_1)$, $E(L_2)$, and f_R are determined by (134), (149), and (174), respectively. In particular, for $P_s = 0$, it holds that $E(S) = E(S_D) \approx (1/2)\,C\,(m-1)\,(m-2)\,(\lambda/\mu)^2\,f_R$.

Remark 42: When P_s increases and approaches 1, from (134), (141), (149), (190), (191), (192), (218), (219), (220), (221), (252), and (255), it follows that $E(S)$ approaches $C\,m$. This is intuitively obvious because when $P_s = 1$, all $C\,m$ symbols stored in the system are lost owing to unrecoverable errors.

We now proceed to derive $E(Q)$, the expected amount of lost user data. First, we note that the expected number of lost user symbols is equal to the product of the storage efficiency to the expected number of lost symbols. Consequently, it follows from (1) that

$$E(Q) = \frac{l}{m}\,E(S)\,s \overset{(3)}{=} \frac{l}{m}\,\frac{E(S)}{C}\,c \,, \qquad (258)$$

where $E(S)$ is given by (255) and s denotes the symbol size.

Similar expressions for the expected amounts $E(Q_{DF_3})$ and $E(Q_{UF})$ of lost user data due to device and unrecoverable failures are obtained from $E(S_D)$ and $E(S_U)$, respectively, as follows:

$$E(Q_{DF_3}) = \frac{l}{m}\,E(S_D)\,s \overset{(3)}{=} \frac{l}{m}\,\frac{E(S_D)}{C}\,c \qquad (259)$$

$$\overset{(222)}{\approx} \frac{l}{m}\,\frac{(m-1)(m-2)}{2}\,\left(\frac{\lambda}{\mu}\right)^2\,f_R\,c \qquad (260)$$

and

$$E(Q_{UF}) = \frac{l}{m}\,E(S_U)\,s \overset{(3)}{=} \frac{l}{m}\,\frac{E(S_U)}{C}\,c \qquad (261)$$

$$\overset{(254)}{\approx} \begin{cases} 3\,\frac{l}{m}\,\frac{(m-1)(m-2)}{2}\,\frac{\lambda}{\mu}\,c\,P_s\,, & \text{for } P_s \ll P_{s^*}^{(5)} \\ 3\,\frac{l}{m}\,\frac{(m-1)(m-2)}{2}\,c\,P_s^2\,, & \text{for } P_{s^*}^{(5)} \ll P_s \ll \frac{1}{m-1}, \end{cases}$$
$$\qquad (262)$$

where $E(S_D)$, $E(S_U)$, and $P_{s^*}^{(5)}$ are determined by (218), (252), and (198), respectively.

Substituting (256) and (257) into (258) yields

$$E(Q) \approx \frac{l}{m}\left[E(L_1) + \frac{1}{2}\,E(L_2)\,(m-1)\,\frac{\lambda}{\mu} \right.$$
$$\left. + \frac{(m-1)(m-2)}{2}\,\left(\frac{\lambda}{\mu}\right)^2\,f_R \right]c \qquad (263)$$

$$\approx \frac{l}{m}\,\frac{(m-1)(m-2)}{2}\left[\left(\frac{\lambda}{\mu}\right)^2 f_R + 3\,\frac{\lambda}{\mu}\,P_s + 3\,P_s^2 \right]c,$$
$$\text{for } P_s \ll \frac{1}{m-1}\,, \qquad (264)$$

where $E(L_1)$, $E(L_2)$, and f_R are obtained by (134), (149), and (174), respectively. In particular, for $P_s = 0$, it holds that $E(Q) = E(Q_{DF_3})$, which is determined by (260).

From (255), (258), (259), and (261), it holds that

$$E(Q) = E(Q_{DF_3}) + E(Q_{UF}) \,. \qquad (265)$$

Also, the expected amounts $E(Q_{UF,1})$, $E(Q_{UF,2})$, and $E(Q_{UF,3})$ of lost user data due to unrecoverable failures in conjunction with one, two, and three device failures are as follows:

$$E(Q_{UF,1}) = \frac{l}{m}\,\frac{E(S_{U,1})}{C}\,c \,, \qquad (266)$$

$$E(Q_{UF,2}) = \frac{l}{m}\,\frac{E(S_{U,2})}{C}\,c \,, \qquad (267)$$

and

$$E(Q_{UF,3}) = \frac{l}{m}\,\frac{E(S_{U,3})}{C}\,c \,, \qquad (268)$$

where $E(S_{U,1})$, $E(S_{U,2})$, and $E(S_{U,3})$ are determined by (141), (192), and (223), respectively.

Remark 43: From (253), (261), (266), and (267), it follows that

$$E(Q_{UF}) \approx \begin{cases} E(Q_{UF,2})\,, & \text{for } P_s \ll P_{s^*}^{(5)} \\ E(Q_{UF,1})\,, & \text{for } P_s \gg P_{s^*}^{(5)}. \end{cases} \qquad (269)$$

Remark 44: Note that the inequalities derived in Remark 34 together with (224) and (225), and by virtue of (259) and (261), imply that

$$E(Q_{\mathrm{UF}}) \ll E(Q_{\mathrm{DF}_3}) \;\Leftrightarrow\; P_s \ll P_{s^*}^{(5)}\,, \qquad (270)$$

where $P_{s^*}^{(5)}$ is determined by (198).

Also, from (260), (262), (269), and (270), it follows that

$$E(Q) \approx \begin{cases} E(Q_{\mathrm{DF}_3})\,, & \text{for } P_s \ll P_{s^*}^{(5)} \\ E(Q_{\mathrm{UF}})\,, & \text{for } P_s \gg P_{s^*}^{(5)} \end{cases} \qquad (271)$$

$$\approx \begin{cases} \frac{l}{m}\,\frac{(m-1)(m-2)}{2}\left(\frac{\lambda}{\mu}\right)^2 f_R\, c\,, & \text{for } P_s \ll P_{s^*}^{(5)} \\ 3\,\frac{l}{m}\,\frac{(m-1)(m-2)}{2}\, c\, P_s^2\,, & \text{for } P_{s^*}^{(5)} \ll P_s \ll \frac{1}{m-1}\,, \end{cases} \qquad (272)$$

where $P_{s^*}^{(5)}$ is determined by (198).

Remark 45: When P_s increases and approaches 1, from (258) and according to Remark 42, it follows that $E(Q)$ approaches $c\,l$. This is intuitively obvious because when $P_s = 1$, upon the first device failure, the entire amount $c\,l$ of user data stored in the RAID-6 array is lost owing to unrecoverable errors.

F. Reliability Metrics

The MTTDL is obtained by substituting (229) into (9). From (229), it follows that MTTDL is insensitive to device failure distribution, but it depends on the rebuild time distribution through f_R and on their means $1/\lambda$ and $1/\mu$, respectively. In particular, the normalized MTTDL depends only on f_R and the ratio λ/μ of their means. Note that for $P_s = 0$, $n = m = N$, and for an exponential rebuild time distribution, for which it holds that $f_R = E(R^2)/[E(R)]^2 = 2$, (231) implies that

$$\mathrm{MTTDL} \approx \frac{\mu^2}{N(N-1)(N-2)\,\lambda^3}\,, \qquad (273)$$

which is the same result as that reported in [3]. Also, for small values of P_s, (231) yields

$$\mathrm{MTTDL} \approx \frac{\mu^2}{N(N-1)(N-2)\,\lambda^2\left(\lambda + \frac{1}{2}\,\mu\, C\, P_s\right)}\,, \qquad (274)$$

whereas Equation (110) of [11] yields

$$\mathrm{MTTDL} \approx \frac{\mu^2}{N(N-1)(N-2)\,\lambda^2\left(\lambda + \mu\, C\, P_s\right)}\,. \qquad (275)$$

Their difference is the factor 1/2, which is attributed to the fact that the probability $P_{\mathrm{uf}}^{(2)}$ of data loss due to an unrecoverable failure given two device failures is obtained in [11] by assuming that all C codewords are to be recovered. It is subsequently obtained in [11] by expression (94) and is equal to $(N-2)\,C\,P_s$. This measure corresponds to $P_{\mathrm{UF}|2}$ whose value, according to (169), is roughly equal to $1/2\,(N-2)\,C\,P_s$, which is half that of $P_{\mathrm{uf}}^{(2)}$. This is because, after the second device failure and according to (31), the expected number of codewords to be recovered in the critical mode is only half the total of codewords C.

The EAFDL is obtained by substituting (258) into (10). In particular, the EAFDL normalized to λ is obtained by substituting (272) into (10) as follows:

$$\mathrm{EAFDL}/\lambda$$

$$\approx \begin{cases} \frac{1}{2}\,(m-1)(m-2)\left(\frac{\lambda}{\mu}\right)^2 f_R\,, & \text{for } P_s \ll P_{s^*}^{(5)} \\ \frac{3}{2}\,(m-1)(m-2)\, P_s^2\,, & \text{for } P_{s^*}^{(5)} \ll P_s \ll \frac{1}{m-1}\,, \end{cases} \qquad (276)$$

where λ/μ and $P_{s^*}^{(5)}$ are determined by (5) and (198), respectively. Note that EAFDL is insensitive to the device failure distribution, but it depends on the rebuild time distribution through f_R and on their means $1/\lambda$ and $1/\mu$, respectively. In particular, the normalized EAFDL depends only on f_R and the ratio λ/μ of their means. Also, for $P_s = 0$, and according to (276), we obtain $\mathrm{EAFDL}/\lambda \approx (1/2)\,(m-1)\,(m-2)\,(\lambda/\mu)^2 f_R$, which is in agreement with Equation (74) of [14] (with $c/b = 1/\mu$ and $\phi = 1$).

The value of $E(H)$ is obtained by substituting (229) and (258) into (11). In particular, depending on the values of λ/μ, m and C and for the two cases considered in Section VI-D, $E(H)$ normalized to c is obtained by substituting (244) or (245) and (272) into (11) as follows:

Case 1: $\frac{\lambda}{\mu} \geq \frac{1}{C}\cdot\frac{2}{m-2} = P_{s^*}^{(2)}$. Depending on the value of λ/μ, we consider two subcases:

(a) $\frac{\lambda}{\mu} \geq \sqrt{\frac{2}{C(m-1)(m-2)}}$. In this case it holds that

$$P_{s^*}^{(1)} < P_{s^*}^{(2)} < P_{s^*}^{(3)} < P_{s^*}^{(4)} \leq P_{s^*}^{(5)}\,. \qquad (277)$$

(b) $\frac{1}{C}\cdot\frac{2}{m-2} \leq \frac{\lambda}{\mu} < \sqrt{\frac{2}{C(m-1)(m-2)}}$. In this case it holds that

$$P_{s^*}^{(1)} < P_{s^*}^{(2)} < P_{s^*}^{(3)} \leq P_{s^*}^{(5)} < P_{s^*}^{(4)}\,. \qquad (278)$$

Then, the $E(H)$ normalized to c is obtained by substituting (244) and (272) into (11) as follows:

$$E(H)/c$$

$$\approx \begin{cases} \frac{l}{m}\,, & \text{for } P_s \ll P_{s^*}^{(1)} \\ \frac{l}{m}\,\frac{\lambda}{\mu}\, f_R\,\frac{1}{C\, P_s}\,, & \text{for } P_{s^*}^{(1)} \ll P_s \ll P_{s^*}^{(2)} \\ \frac{l}{m}\,\frac{m-2}{2}\,\frac{\lambda}{\mu}\, f_R\,, & \text{for } P_{s^*}^{(2)} \ll P_s \ll P_{s^*}^{(3)} \\ \frac{l}{m}\,\left(\frac{\lambda}{\mu}\right)^2 f_R\,\frac{1}{C\, P_s^2}\,, & \text{for } P_{s^*}^{(3)} \ll P_s \ll P_{\min}^{(4,5)} \\ \frac{l}{m}\,\max\left(\frac{(m-1)(m-2)}{2}\left(\frac{\lambda}{\mu}\right)^2 f_R\,,\,\frac{3}{C}\right)\,, & \\ & \text{for } P_{\min}^{(4,5)} \ll P_s \ll P_{\max}^{(4,5)} \\ 3\,\frac{l}{m}\,\frac{(m-1)(m-2)}{2}\, P_s^2\,, & \text{for } P_{\max}^{(4,5)} \ll P_s \ll \frac{1}{m-1}\,, \end{cases} \qquad (279)$$

where

$$P_{\min}^{(4,5)} \triangleq \min(P_{s^*}^{(4)}, P_{s^*}^{(5)}) \quad \text{and} \quad P_{\max}^{(4,5)} \triangleq \max(P_{s^*}^{(4)}, P_{s^*}^{(5)})\,, \qquad (280)$$

and $P_{s^*}^{(1)}$, $P_{s^*}^{(2)}$, $P_{s^*}^{(3)}$, $P_{s^*}^{(4)}$, and $P_{s^*}^{(5)}$ are determined by (233), (162), (237), (131), (198), respectively. Note that $E(H)$, as a function of P_s, exhibits three plateaus in the intervals $[0, P_{s^*}^{(1)})$, $(P_{s^*}^{(2)}, P_{s^*}^{(3)})$, and $(P_{\min}^{(4,5)}, P_{\max}^{(4,5)})$, respectively.

Case 2: $\frac{\lambda}{\mu} < \frac{1}{C}\cdot\frac{2}{m-2} = P_{s^*}^{(2)}$. In this case it holds that

$$P_{s^*}^{(1)} < P_{s^*}^{(3)} = P_{s^*}^{(5)} < P_{s^*}^{(2)} < P_{s^*}^{(4)}\,. \qquad (281)$$

The value of $E(H)$ normalized to c is obtained by substituting (245) and (272) into (11) as follows:

$$\frac{E(H)}{c} \approx \begin{cases} \frac{l}{m} , & \text{for } P_s \ll P_s^{(1)} \\ \frac{l}{m} \frac{\lambda}{\mu} f_R \frac{1}{C P_s} , & \text{for } P_{s*}^{(1)} \ll P_s \ll P_{s*}^{(5)} \\ 3 \frac{l}{m} \frac{1}{C} , & \text{for } P_{s*}^{(5)} \ll P_s \ll P_{s*}^{(4)} \\ 3 \frac{l}{m} \frac{(m-1)(m-2)}{2} P_s^2 , & \text{for } P_{s*}^{(4)} \ll P_s \ll \frac{1}{m-1} , \end{cases}$$
$$(282)$$

where $P_{s*}^{(1)}$, $P_{s*}^{(4)}$, and $P_{s*}^{(5)}$ are determined by (233), (131), and (198), respectively. Note that $E(H)$, as a function of P_s, exhibits two plateaus in the intervals $[0, P_{s*}^{(1)})$ and $(P_{s*}^{(5)}, P_{s*}^{(4)})$, respectively.

Analogous to (119), the expected amounts $E(H_{\mathrm{DF}_3})$ and $E(H_{\mathrm{UF}})$ of user data lost due to device and unrecoverable failures, given that such failures have occurred, are

$$E(H_{\mathrm{DF}_3}) = \frac{E(Q_{\mathrm{DF}_3})}{P_{\mathrm{DF}_3}}, \quad \text{and} \quad E(H_{\mathrm{UF}}) = \frac{E(Q_{\mathrm{UF}})}{P_{\mathrm{UF}}},$$
$$(283)$$

respectively. Also, analogous to (120), the relation between $E(H)$, $E(H_{\mathrm{DF}_3})$, and $E(H_{\mathrm{UF}})$ is

$$E(H) = \frac{P_{\mathrm{DF}_3}}{P_{\mathrm{DL}}} E(H_{\mathrm{DF}_3}) + \frac{P_{\mathrm{UF}}}{P_{\mathrm{DL}}} E(H_{\mathrm{UF}}) .$$
$$(284)$$

Substituting (205) and (260) into (283) yields

$$E(H_{\mathrm{DF}_3})/c \approx \frac{l}{m} .$$
$$(285)$$

Also, depending on the values of λ/μ, m and C and for the two cases considered in Section VI-D, the $E(H_{\mathrm{UF}})$ normalized to c is obtained by substituting (250) or (251), and (262) into (283) as follows:

Case 1: $\frac{\lambda}{\mu} \geq \frac{1}{C} \cdot \frac{2}{m-2} = P_{s*}^{(2)}$.

$E(H_{\mathrm{UF}})/c$

$$\approx \begin{cases} 3 \frac{l}{m} \frac{1}{C} , & \text{for } P_s \ll P_{s*}^{(2)} \\ 3 \frac{l}{m} \frac{m-2}{2} P_s , & \text{for } P_{s*}^{(2)} \ll P_s \ll P_{s*}^{(3)} \\ 3 \frac{l}{m} \frac{\lambda}{\mu} \frac{1}{C P_s} , & \text{for } P_{s*}^{(3)} \ll P_s \ll P_{\min}^{(4,5)} \\ 3 \frac{l}{m} \max\left(\frac{(m-1)(m-2)}{2} \frac{\lambda}{\mu} P_s , \frac{1}{C} \right) , & \\ & \text{for } P_{\min}^{(4,5)} \ll P_s \ll P_{\max}^{(4,5)} \\ 3 \frac{l}{m} \frac{(m-1)(m-2)}{2} P_s^2 , & \text{for } P_{\max}^{(4,5)} \ll P_s \ll \frac{1}{m-1} , \end{cases}$$
$$(286)$$

where $P_{s*}^{(1)}$, $P_{s*}^{(2)}$, $P_{s*}^{(3)}$, $P_{\min}^{(4,5)}$ and $P_{\max}^{(4,5)}$ are determined by (233), (162), (237), and (280), respectively. Note that $E(H_{\mathrm{UF}})$ generally increases with P_s, but in the interval $(P_{s*}^{(3)}, P_{\min}^{(4,5)})$ decreases with P_s.

Case 2: $\frac{\lambda}{\mu} < \frac{1}{C} \cdot \frac{2}{m-2} = P_{s*}^{(2)}$.

$$\frac{E(H_{\mathrm{UF}})}{c} \approx \begin{cases} 3 \frac{l}{m} \frac{1}{C} , & \text{for } P_s \ll P_{s*}^{(4)} \\ 3 \frac{l}{m} \frac{(m-1)(m-2)}{2} P_s^2 , & \text{for } P_{s*}^{(4)} \ll P_s \ll \frac{1}{m-1} , \end{cases}$$
$$(287)$$

where $P_{s*}^{(4)}$ is determined by (131).

VII. NUMERICAL RESULTS

A. A RAID-5 System

We consider a RAID-5 array comprised of $n = 8$ devices with $N = m = 8$, $l = 7$, $\lambda/\mu = 0.001$, capacity $c = 1\mathrm{TB}$, and symbol size s equal to a sector size of 512 bytes, such that the number of codewords stored in a device is $C = c/s = 1.9 \times 10^9$.

The probability of data loss P_{DL} is determined by (83) as a function of the unrecoverable error probability P_s of a symbol (sector), and shown in Figure 2. According to (33), the probability P_{DF_2} of a device failure occurring during rebuild is independent of the unrecoverable symbol error probability, as indicated by the horizontal dotted blue line in Figure 2. It follows from (82) and (87) that, for $P_s \ll P_s^{(1)}$, an unrecoverable failure most likely occurs in the case of one device failure with the corresponding probability $P_{\mathrm{UF},1}$ being much smaller than the probability P_{DF_2} of encountering a device failure during rebuild, as shown in Figure 2. However, when P_s is in the range $(P_s^{(1)}, P_s^{(2)})$, $P_{\mathrm{UF},1}$ becomes greater than P_{DF_2}, which implies that a data loss is most likely caused by an unrecoverable failure that occurs in the case of one device failure. In particular, for $P_s \ll P_s^{(2)}$, and according to (39), $P_{\mathrm{UF},1}$ increases linearly with P_s, as indicated by the dotted green line in Figure 2. It follows from (88) and (21), and for the parameters considered here, that $P_s^{(1)} = 5 \times 10^{-13}$ and $P_s^{(2)} = 7 \times 10^{-11}$, as shown in Figure 2. For $P_s \gg P_s^{(2)}$, and according to (89), (91), and (92), it follows that $P_{\mathrm{UF},1}$, P_{UF} and, in turn, P_{DL} approach 1 and are essentially independent of P_s. In this range and according to (19), a device failure leads to data loss because one of the codewords is almost surely corrupted. Note that this also holds in the case when a subsequent (second) device failure occurs during rebuild. Consequently, and according to (66), $P_{\mathrm{UF},2}$ approaches P_{DF_2}, as indicated by the dotted magenta line in Figure 2. As expected, and according to (86), the total probability of data loss P_{DL} increases monotonically with P_s and exhibits two plateaus in the intervals $[0, P_s^{(1)})$ and $(P_s^{(2)}, 1]$, respectively.

The normalized λ MTTDL measure is obtained from (113) and is shown in Figure 3 as a function of the unrecoverable symbol error probability. The various regions and plateaus are also depicted and correspond to the ranges discussed above regarding the probability of data loss.

The normalized expected amount $E(Q)/c$ of lost user data relative to the amount of data stored in a device is obtained from (99) as a function of the unrecoverable symbol error probability P_s, and shown in Figure 4. According to (101), the normalized expected amount $E(Q_{\mathrm{DF}_2})/c$ of user data lost due to a subsequent device failure during rebuild is independent of the unrecoverable symbol error probability, as indicated by the horizontal dotted blue line in Figure 4. The normalized expected amount $E(Q_{\mathrm{UF}})/c$ of user data lost due to unrecoverable failures, and according to (109), is roughly equal to $E(Q_{\mathrm{UF},1})/c$, which is determined by (107) and shown in Figure 4. In particular, for small values of P_s, and according to (103), it increases linearly with P_s, as indicated by the dotted green line in Figure 4. Also, the expected amount $E(Q_{\mathrm{UF},2})$ of user data lost due to unrecoverable failures in conjunction with two device failures, and according to Remark

Figure 2. Probability of data loss P_{DL} for a RAID-5 array with latent errors ($\lambda/\mu = 0.001$, $m = N = 8$, $l = 7$, $c = 1$TB, and $s = 512$ B).

Figure 4. Normalized amount of data loss $E(Q)$ for a RAID-5 array with latent errors ($\lambda/\mu = 0.001$, $m = N = 8$, $l = 7$, $c = 1$TB, and $s = 512$ B).
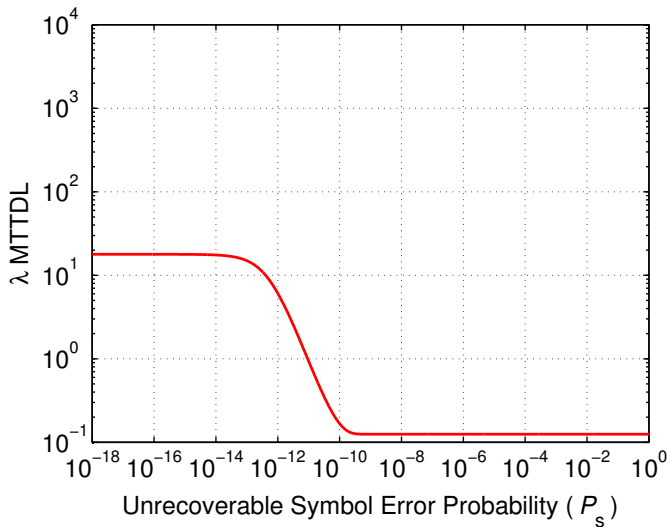
Figure 3. Normalized MTTDL for a RAID-5 array with latent errors ($\lambda/\mu = 0.001$, $m = N = 8$, $l = 7$, $c = 1$TB, and $s = 512$ B).
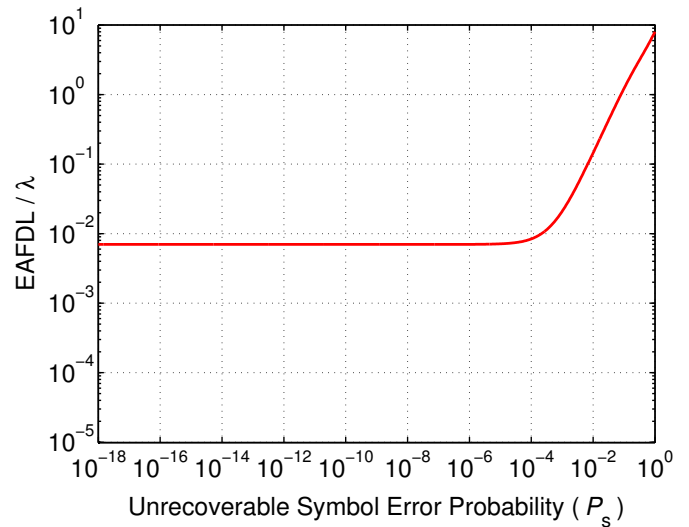
Figure 5. Normalized EAFDL for a RAID-5 array with latent errors ($\lambda/\mu = 0.001$, $m = N = 8$, $l = 7$, $c = 1$TB, and $s = 512$ B).

6, is much smaller than $E(Q_{\mathrm{UF},1})$, as indicated by the dotted magenta line in Figure 4. According to (110), $E(Q_{\mathrm{UF}})$ exceeds $E(Q_{\mathrm{DF}_2})$ when $P_s \gg P_s^{(3)} = 5{\times}10^{-4}$. As expected, and according to (105), the total expected amount $E(Q)$ of lost user data increases monotonically with P_s. In particular, when P_s approaches 1 and according to Remark 14, the normalized expected amount $E(Q)/c$ of lost user data approaches $l = 7$, as all user data in the array is lost.

The normalized EAFDL/λ measure is obtained by substituting (99) into (10) and is shown in Figure 5 as a function of the unrecoverable symbol error probability. Equation (10) suggests that this measure is proportional to $E(Q)$, which implies that the above discussion regarding the behavior of $E(Q)$ also holds here and therefore EAFDL increases monotonically with P_s.

The normalized expected amount $E(H)/c$ of lost user data, given that a data loss has occurred, relative to the amount of data stored in a device is obtained from (118) as a function of the unrecoverable symbol error probability P_s and shown in Figure 6. In contrast to the P_{DL}, EAFDL, and $E(Q)$ measures that increase monotonically with P_s, we observe that $E(H)$ does not do so. Data losses occur because of a subsequent device failure or unrecoverable failures of codewords, or a combination thereof. According to (123), the expected amount $E(H_{\mathrm{DF}_2})$ of lost user data associated with a subsequent device failure, given that such a device failure has occurred during rebuild, is independent of the unrecoverable symbol error probability, as indicated by the horizontal dotted blue line in Figure 6. Such a device failure causes the loss of a large number of symbols as opposed to a small number of additional symbols that may be lost owing to unrecoverable failures. The expected amount $E(H_{\mathrm{UF}})$ of user data lost due to
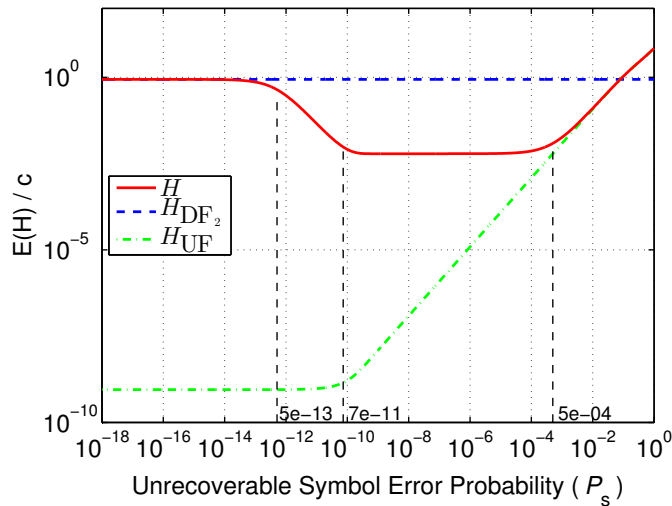
Figure 6. Normalized $E(H)$ for a RAID-5 array with latent errors ($\lambda/\mu = 0.001$, $m = N = 8$, $l = 7$, $c = 1$TB, and $s = 512$ B).
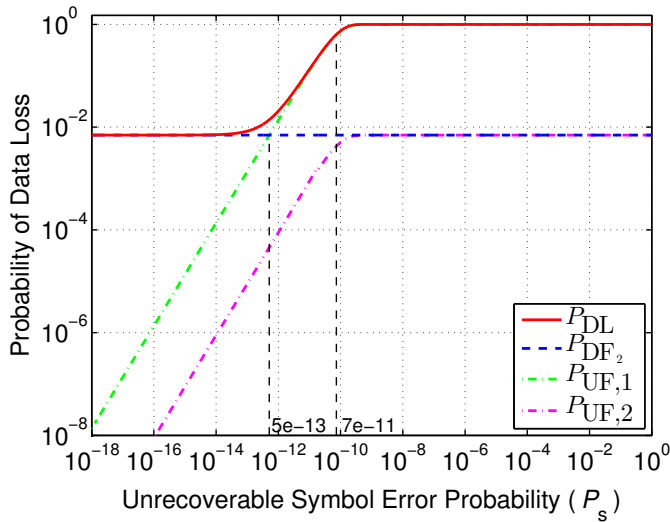

Figure 7. Probability of data loss P_{DL} for a RAID-6 array with latent errors ($\lambda/\mu = 0.001$, $f_R = 2$, $m = N = 8$, $l = 6$, $c = 1$TB, and $s = 512$ B).

unrecoverable failures, given that such failures have occurred, is obtained from (124) and shown in Figure 6. According to Remark 1, (24), and (28), when $P_s \ll P_s^{(2)} = 7 \times 10^{-11}$, an unrecoverable failure is most likely caused by a single corrupted codeword that loses two symbols. Consequently, and according to (125), for $P_s \ll P_s^{(2)}$, the expected amount $E(H_{\text{UF}})$ of user data lost due to unrecoverable failures, given that such unrecoverable failures have occurred, is independent of P_s, as indicated by the horizontal part of the dotted green line in Figure 6. Also, the amount of lost data, which corresponds to the two lost symbols, is negligible compared with the amount of data lost due to a subsequent device failure, that is, $E(H_{\text{UF}}) \ll E(H_{\text{DF}_2})$. According to (24) and (28), when $P_s \gg P_s^{(2)}$, unrecoverable failures are most likely caused by multiple corrupted codewords that each loses two symbols. Moreover, (24) and (125) imply that the number of the corrupted codewords and the corresponding amount of lost data increase linearly with P_s, as indicated by the dotted green line shown in Figure 6.

The combined expected amount $E(H)$ of lost user data, given that data loss has occurred, is an average of $E(H_{\text{DF}_2})$ and $E(H_{\text{UF}})$ with the weights determined in (120). For $P_s \ll P_s^{(1)} = 5 \times 10^{-13}$, a data loss is most likely attributed to two device failures, which results in the first plateau obtained in (123). However, for values of P_s in the range (5×10^{-13}, 7×10^{-11}), this is reversed, meaning that an unrecoverable failure is more likely to occur than a device failure, and this causes P_{DL} to increase as shown in Figure 2. Consequently, as the weight of the $E(H_{\text{DF}_2})$ component decreases, so does $E(H)$. Subsequently, as P_s increases further, this weight along with $E(H)$ can no longer decrease because P_{DL} has reached its maximum value of 1. But, $E(H)$ cannot increase either because, although $E(H_{\text{UF}})$ increases, it still remains negligible compared with $E(H_{\text{DF}_2})$. As a result, $E(H)$ stabilizes at the second plateau level at $(l/m)(m-1)(\lambda/\mu)c$, as obtained by (121). As P_s increases further and exceeds $P_s^{(3)} = 5 \times 10^{-4}$, according to (110), the increasing amount of data lost due to unrecoverable failures $E(Q_{\text{UF}})$ far exceeds $E(Q_{\text{DF}_2})$, which in

turn leads $E(H)$ to be essentially equal to $E(H_{\text{UF}})$ and therefore to increase with P_s. In particular, when P_s approaches 1, and according to Remark 17, the amount $l\,c$ of user data stored in the array is lost owing to unrecoverable errors, which in turn implies that the normalized expected amount $E(H)/c$ of lost user data approaches $l = 7$.

B. A RAID-6 System

We consider a RAID-6 array with the same characteristics as the RAID-5 array considered in the previous section, except that the parameter l is now equal to 6. Also, in contrast to a RAID-5 system, some of the reliability metrics for a RAID-6 system depend on the rebuild time distribution. We consider a rebuild time distribution, such as the exponential one, for which it holds that $E(R^2) = 2\,[E(R)]^2$, which implies that $f_R = 2$.

The probability of data loss P_{DL} is determined by (229) as a function of the unrecoverable error probability P_s of a symbol (sector), and shown in Figure 7. According to (204), the probability P_{DF_3} of two device failures occurring during rebuild is independent of the unrecoverable symbol error probability, as indicated by the horizontal dotted blue line in Figure 7. It follows from (230) and (232) that, for $P_s \ll P_{s*}^{(1)}$, an unrecoverable failure most likely occurs in conjunction with two device failures with the corresponding probability $P_{\text{UF},2}$ being much smaller than the probability P_{DF_3} of three device failures, as shown in Figure 7. However, when P_s is in the range ($P_{s*}^{(1)}$, $P_{s*}^{(2)}$), $P_{\text{UF},2}$ becomes greater than P_{DF_3}, which implies that data loss is most likely caused by an unrecoverable failure that occurs in conjunction with two device failures. In particular, for $P_s \ll P_{s*}^{(2)}$, and according to (179), $P_{\text{UF},2}$ increases linearly with P_s, as indicated by the dotted cyan line in Figure 7. It follows from (233) and (162), and for the parameters considered here, that $P_{s*}^{(1)} = 1 \times 10^{-12}$ and $P_{s*}^{(2)} = 2 \times 10^{-10}$, as shown in Figure 7. It follows from (230) and (232) that, for $P_s \gg P_{s*}^{(1)}$, the probability P_{UF} of encountering an unrecoverable failure is much greater than that

Figure 8. Normalized MTTDL for a RAID-6 array with latent errors ($\lambda/\mu =$ 0.001, $f_R = 2$, $m = N = 8$, $l = 6$, $c = $ 1TB, and $s = 512$ B).
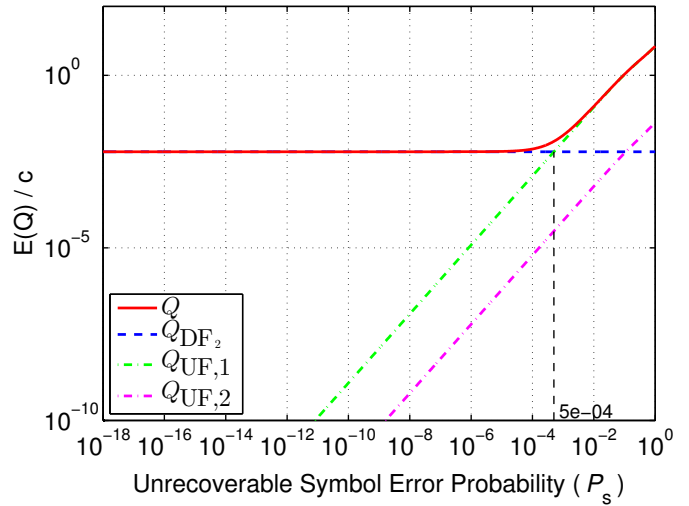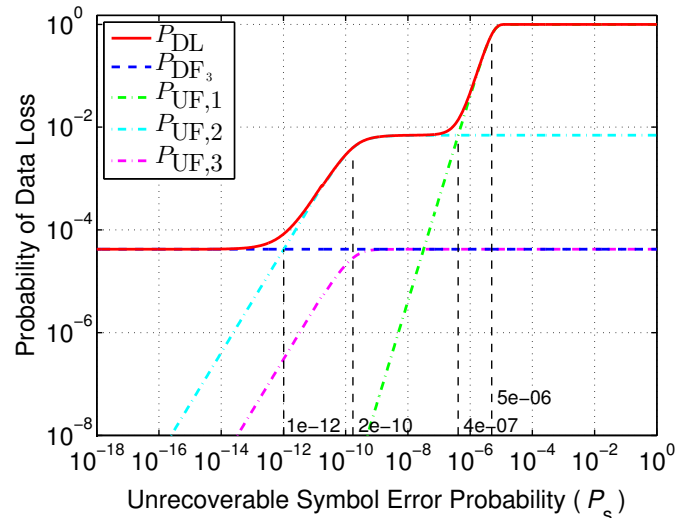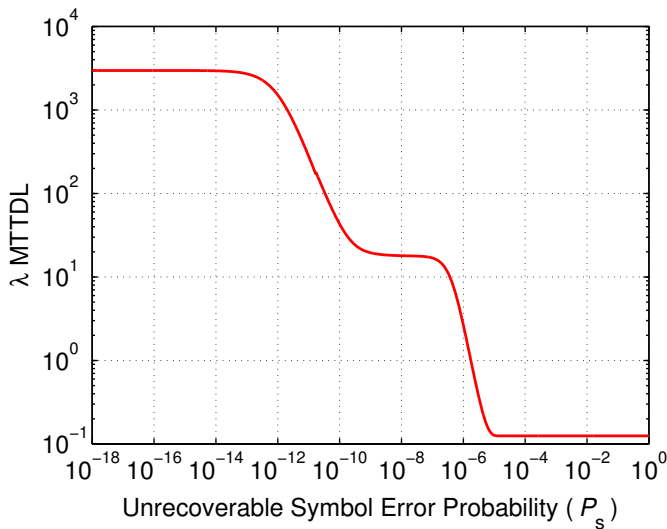

Figure 9. Normalized amount of data loss $E(Q)$ for a RAID-6 array with latent errors ($\lambda/\mu = 0.001$, $f_R = 2$, $m = N = 8$, $l = 6$, $c = $ 1TB, and $s = 512$ B).

of encountering a data loss due to three device failures. In particular, from (169), (177), and (179), it follows that, when $P_s \gg P_{s*}^{(2)}$, $P_{\text{UF}|2}$ approaches 1 and, in turn, $P_{\text{UF,2}}$ approaches P_{DF_2} and they are essentially independent of P_s. In this range and according to Remark 38, a second device failure leads to data loss because one of the remaining codewords is almost surely corrupted, which implies that the probability of data loss is equal to that of a RAID-5 system in the absence of latent errors. Note that this also holds in the case when a subsequent (third) device failure occurs during rebuild. Consequently, and according to (210), $P_{\text{UF,3}}$ approaches P_{DF_3}, as indicated by the dotted magenta line in Figure 7.

Subsequently, according to Remark 39, when $P_s \gg P_{s*}^{(3)}$, the probability $P_{\text{UF,1}}$ of data loss due to unrecoverable failures in the case of one device failure becomes greater than $P_{\text{UF,2}}$, which implies that a data loss is most likely caused by an unrecoverable failure in conjunction with one device failure. In particular, for $P_s \ll P_{s*}^{(4)}$, and according to (138) and (129), $P_{\text{UF,1}}$ increases quadratically with P_s, as indicated by the dotted green line in Figure 7. It follows from (81) and (131), and for the parameters considered here, that $P_{s*}^{(3)} = 4 \times 10^{-7}$ and $P_{s*}^{(4)} = 5 \times 10^{-6}$, as shown in Figure 7. For $P_s \gg P_{s*}^{(4)}$, and according to (187), (244), (247), and (250), it follows that $P_{\text{UF,1}}$, P_{UF} and, in turn, P_{DL} approach 1 and are essentially independent of P_s. In this range and according to (129), a device failure leads to data loss because one of the codewords is almost surely corrupted. As expected, the total probability of data loss P_{DL} increases monotonically with P_s and exhibits three plateaus in the intervals $[0, P_{s*}^{(1)})$, $(P_{s*}^{(2)}, P_{s*}^{(3)})$, and $(P_{s*}^{(4)}, 1]$, respectively.

The normalized $\lambda\,\text{MTTDL}$ measure is obtained by substituting (229) into (9) and is shown in Figure 8 as a function of the unrecoverable symbol error probability. The various regions and plateaus are also depicted and correspond to the ranges discussed above regarding the probability of data loss.

The normalized expected amount $E(Q)/c$ of lost user data

relative to the amount of data stored in a device is obtained from (258) as a function of the unrecoverable symbol error probability P_s, and shown in Figure 9. According to (260), the normalized expected amount $E(Q_{\text{DF}_3})/c$ of user data lost due to two subsequent device failures during rebuild is independent of the unrecoverable symbol error probability, as indicated by the horizontal dotted blue line in Figure 9. The normalized expected amount $E(Q_{\text{UF}})/c$ of user data lost due to unrecoverable failures, when $P_s \ll P_{s*}^{(5)} = 10^{-3}$ and according to (262) and (269), is roughly equal to $E(Q_{\text{UF,2}})/c$ and increases linearly with P_s, as indicated by the dotted cyan line in Figure 9. For $P_s \gg P_{s*}^{(5)} = 10^{-3}$, $E(Q_{\text{UF}})/c$ is roughly equal to $E(Q_{\text{UF,1}})/c$ and increases quadratically with P_s, as indicated by the dotted green line in Figure 9. Also, the expected amount $E(Q_{\text{UF,3}})$ of user data lost due to unrecoverable failures in conjunction with three device failures, and according to Remark 34, is much smaller than $E(Q_{\text{UF,2}})$, as indicated by the dotted magenta line in Figure 9. According to (270), $E(Q_{\text{UF}})$ exceeds $E(Q_{\text{DF}_3})$ when $P_s \gg P_{s*}^{(5)} = 10^{-3}$. As expected, the total expected amount $E(Q)$ of lost user data increases monotonically with P_s. In particular, when P_s approaches 1 and according to Remark 12, the normalized expected amount $E(Q)/c$ of lost user data approaches $l = 6$, as all user data in the array is lost.

The normalized EAFDL/λ measure is obtained by substituting (258) into (10) and is shown in Figure 10 as a function of the unrecoverable symbol error probability. Equation (10) suggests that this measure is proportional to $E(Q)$, which implies that the preceding discussion regarding the behavior of $E(Q)$ also holds here and therefore EAFDL increases monotonically with P_s.

The normalized expected amount $E(H)/c$ of lost user data, given that a data loss has occurred, relative to the amount of data stored in a device is obtained from (279) as a function of the unrecoverable symbol error probability P_s and shown in Figure 11. In contrast to the P_{DL}, EAFDL, and $E(Q)$ measures that increase monotonically with P_s, we observe that $E(H)$

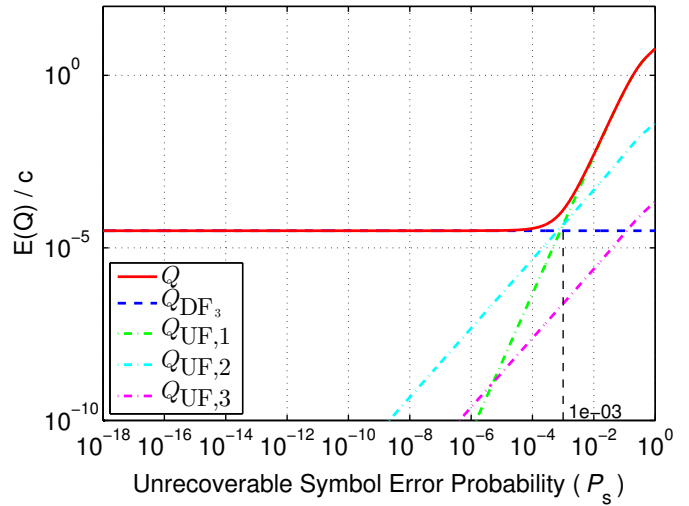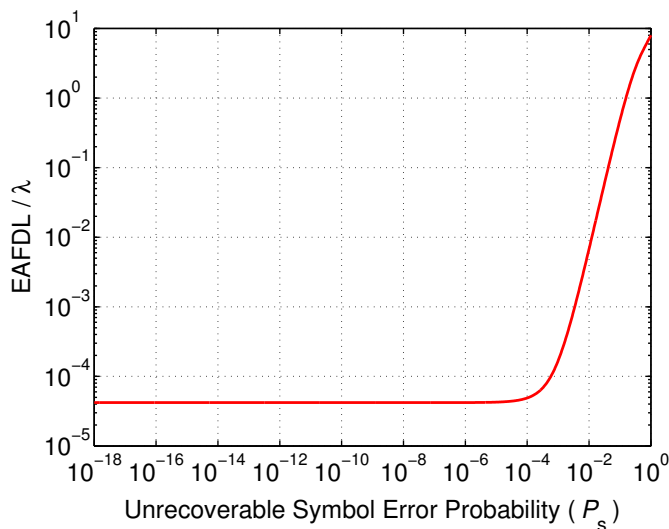Figure 10. Normalized EAFDL for a RAID-6 array with latent errors ($\lambda/\mu = 0.001$, $f_R = 2$, $m = N = 8$, $l = 6$, $c = 1$TB, and $s = 512$ B).

Figure 11. Normalized $E(H)$ for a RAID-6 array with latent errors ($\lambda/\mu = 0.001$, $f_R = 2$, $m = N = 8$, $l = 6$, $c = 1$TB, and $s = 512$ B).

does not do so. Data losses occur because of two subsequent device failures or unrecoverable failures of codewords, or a combination thereof. According to (285), the expected amount $E(H_{\text{DF}_3})$ of lost user data associated with two subsequent device failures, given that such device failures have occurred during rebuild, is independent of the unrecoverable symbol error probability, as indicated by the horizontal dotted blue line in Figure 11. Such device failures cause the loss of a large number of symbols as opposed to a small number of additional symbols that may be lost owing to unrecoverable failures. The expected amount $E(H_{\text{UF}})$ of user data lost due to unrecoverable failures, given that such failures have occurred, is determined by (286) and shown in Figure 11. According to (236), when $P_s \ll P_{s*}^{(3)}$, $P_{\text{UF},2}$ is much greater than $P_{\text{UF},1}$, which implies that an unrecoverable failure most likely occurs in conjunction with two device failures. According to Corollary 11, (151), and (167), when $P_s \ll P_{s*}^{(2)} = 2\times10^{-10}$, an unrecoverable failure is most likely caused by a single corrupted codeword that loses three symbols and is encountered after the second device failure. Consequently, and according to (286), for $P_s \ll P_{s*}^{(2)}$, the expected amount $E(H_{\text{UF}})$ of user data lost due to unrecoverable failures, given that such unrecoverable failures have occurred, is independent of P_s, as indicated by the horizontal part of the dotted green line in Figure 11. Also, the amount of lost data, which corresponds to the three lost symbols, is negligible compared with the amount of data lost due to a subsequent device failure, that is, $E(H_{\text{UF}}) \ll E(H_{\text{DF}_2})$. According to (151) and (167), when $P_s \gg P_s^{(2)}$, unrecoverable failures are most likely caused by multiple corrupted codewords that each loses three symbols and are encountered after the second device failure. Moreover, (167) and (286) imply that the number of corrupted codewords and the corresponding amount of lost data increase linearly with P_s in $(P_{s*}^{(2)}, P_{s*}^{(3)})$, as indicated by the dotted green line in Figure 11. Subsequently, when $P_s \gg P_{s*}^{(3)}$, and according to (133) and (136), unrecoverable failures may also be caused by a single corrupted codeword that is encountered in conjunction with one device failure and loses three symbols. This in turn
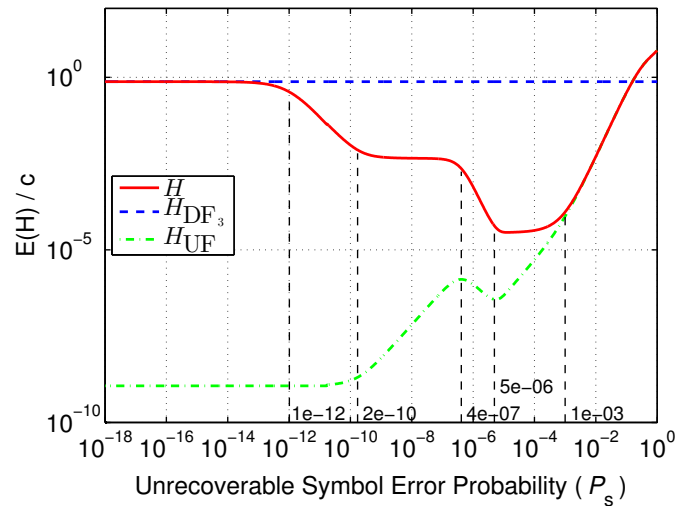
results in a reduction of the amount of lost data, as expressed by (286). When $P_s \gg P_{s*}^{(4)}$, and according to (129), (133), and (136), the expected number of corrupted codewords in the case of one device failure also increases. In particular, unrecoverable failures are more likely encountered in conjunction with one device failure, but although $E(hufo)$ increases, it remains negligible compared with $E(huft)$, which also increases. This in turn results in an increase of the amount of lost data, as expressed by (286).

The combined expected amount $E(H)$ of lost user data, given that data loss has occurred, is an average of $E(H_{\text{DF}_3})$ and $E(H_{\text{UF}})$ with the weights determined in (284). For $P_s \ll P_{s*}^{(1)} = 1\times10^{-12}$, a data loss is most likely attributed to three device failures, which results in the first plateau obtained in (285). However, for higher values of P_s, this is reversed, in that it becomes more likely to encounter an unrecoverable failure than a device failure. As in the case of RAID-5, $E(H)$ initially decreases, but in the case of RAID-6 it exhibits three plateaus, as obtained in (279). As P_s increases further and exceeds $P_{s*}^{(5)} = 10^{-3}$, according to (270), the increasing amount of data lost due to unrecoverable failures $E(Q_{\text{UF}})$ far exceeds $E(Q_{\text{DF}_3})$, which in turn leads $E(H)$ to be essentially equal to $E(H_{\text{UF}})$ and therefore to increase with P_s. In particular, when P_s approaches 1, and according to Remark 45, the amount $l\,c$ of user data stored in the array is lost owing to unrecoverable errors, which in turn implies that the normalized expected amount $E(H)/c$ of lost user data approaches $l = 7$.

VIII. DISCUSSION

As discussed in Section III, field results suggest that the probability of unrecoverable sector errors lies in the range $(4.096\times10^{-11}, 5\times10^{-9})$. Figures 3 and 8 show that MTTDL is significantly degraded by the presence of latent errors, whereas Figures 5 and 10 reveal that EAFDL is practically unaffected in this range. When the probability of unrecoverable sector errors lies in the range of practical interest, the probability of an unrecoverable failure is much greater than that of a data loss due to device failures, which degrades MTTDL. However,

the amount of data corresponding to sectors lost due to latent errors is negligible compared with the amount of data lost due to device failures, which in turn implies that EAFDL remains practically unaffected. In contrast, Figures 6 and 11 reveal that the expected amount $E(H)$ of lost user data, given that data loss has occurred, decreases in the range of practical interest. This is because when a data loss occurs, it is more likely caused by unrecoverable failures that involve the loss of a small number of sectors rather than by multiple device failures that result in a significantly greater amount of lost data.

It follows from (88) and (81) that

$$P_s^{(1)} = \frac{1}{C} \cdot \frac{\lambda}{\mu} \ \ll \ \frac{1}{2} \cdot \frac{C+1}{C} \cdot \frac{\lambda}{\mu} = P_s^{(3)} \ . \qquad (288)$$

Similarly, from (233) and (198), it holds that

$$P_{s^*}^{(1)} = \frac{1}{C} \cdot \frac{\lambda}{\mu} \cdot f_R \ \ll \ \frac{\lambda}{\mu} = P_{s^*}^{(5)} \ . \qquad (289)$$

Consequently, increasing P_s first affects P_{DL}, MTTDL, and $E(H)$ and then $E(Q)$ and EAFDL.

IX. Conclusions

The effect of latent sector errors on the reliability of RAID-5 and RAID-6 data storage systems was investigated. A methodology was developed for deriving the Mean Time to Data Loss (MTTDL) and the Expected Annual Fraction of Data Loss (EAFDL) reliability metrics analytically. Closed-form expressions capturing the effect of unrecoverable latent errors were obtained. Our results demonstrate that RAID-6 storage systems achieve a higher reliability than that of RAID-5 storage systems. We established that the reliability of storage systems is adversely affected by the presence of latent errors. The results demonstrated that the effect of latent errors depends on the relative magnitudes of the probability of a latent error versus the probability of a device failure. It was found that, for actual values of the unrecoverable sector error probability, MTTDL is adversely affected by the presence of latent errors, whereas EAFDL is not.

Extending the methodology developed to derive the MTTDL and EAFDL reliability metrics of erasure-coded systems in the presence of unrecoverable latent errors is a subject of further investigation.

Appendix A

Proof of Corollary 4.

For small values of x, it holds that

$$(1 + x)^n \approx 1 + nx + \frac{n(n-1)}{2} x^2 + \frac{n(n-1)(n-2)}{6} x^3$$
$$+ \frac{n(n-1)(n-2)(n-3)}{24} x^4 \ . \qquad (290)$$

Consequently, for $x = -P_s$ and $n = (m-1)\,C$, (290) yields

$$q_1^C \stackrel{(14)}{=} (1 - P_s)^{(m-1)C}$$
$$\approx 1 - (m-1)C\,P_s + \frac{(m-1)C[(m-1)C-1]}{2}\,P_s^2 \ . \qquad (291)$$

By setting $C = 1$, (291) yields

$$q_1 \approx 1 - (m-1)\,P_s + \frac{(m-1)(m-2)}{2}\,P_s^2 \ . \qquad (292)$$

Substituting (291) and (292) into (56) yields

$$P_{\mathrm{UF \ in} \ S_1|2} \approx 1 - \frac{(m-1)C\,P_s - \frac{(m-1)C[(m-1)C-1]}{2}\,P_s^2}{C\,[(m-1)\,P_s - \frac{(m-1)(m-2)}{2}\,P_s^2]}$$
$$= 1 - \frac{1 - \frac{[(m-1)C-1]}{2}\,P_s}{1 - \frac{m-2}{2}\,P_s} \approx \frac{(C-1)(m-1)}{2}\,P_s \ . \qquad (293)$$

This approximation holds when $(1/2)(C-1)(m-1)\,P_s \ll 1$ or, equivalently, $P_s \ll 2/[(C-1)(m-1)]$, which is roughly twice the value of $P_s^{(2)}$ as given by (21). ∎

Appendix B

Proof of Corollary 5.

For $x = -P_s$ and $n = (m-2)\,C$, (290) yields

$$p_2^C \stackrel{(51)}{=} (1 - P_s)^{(m-2)C}$$
$$\approx 1 - (m-2)C\,P_s + \frac{(m-2)C[(m-2)C-1]}{2}\,P_s^2 \ . \qquad (294)$$

By setting $C = 1$, (294) yields

$$p_2 \approx 1 - (m-2)\,P_s + \frac{(m-2)(m-3)}{2}\,P_s^2 \ . \qquad (295)$$

Substituting (294) and (295) into (61) yields

$$P_{\mathrm{UF \ in} \ S_2|2} \approx 1 - [1 - (m-2)P_s]\,\frac{1 - \frac{C(m-2)-1}{2}\,P_s}{1 - \frac{m-3}{2}\,P_s}$$
$$\approx \frac{\frac{(C+1)(m-2)}{2}\,P_s}{1 - \frac{m-3}{2}\,P_s} \approx \frac{(C+1)(m-2)}{2}\,P_s \ . \qquad (296)$$

This approximation holds when $(1/2)(C+1)(m-2)\,P_s \ll 1$ or, equivalently, $P_s \ll 2/[(C+1)(m-2)]$, which for large C is roughly equal to $2/[C(m-2)]$. ∎

Appendix C

Proof of Corollary 6.

From (14) and (51) it follows that

$$q_1 = (1 - P_s)\,p_2 \ . \qquad (297)$$

Then it holds that

$$\frac{p_2}{C}\,\frac{p_2^C - q_1^C}{p_2 - q_1} \stackrel{(297)}{=} \frac{p_2}{C}\,\frac{[1 - (1-P_s)^C]p_2^C}{P_s\,p_2} = \frac{[1 - (1-P_s)^C]}{C\,P_s}\,p_2^C$$
$$\stackrel{(52)}{\approx} \frac{[1 - (1 - C\,P_s + \frac{C(C-1)}{2}\,P_s^2)]}{C\,P_s}\,[1 - (m-2)\,P_s]^C$$
$$\approx \left[1 - \frac{(C-1)}{2}\,P_s\right][1 - C\,(m-2)\,P_s]$$
$$= 1 - \left[\frac{C-1}{2} + (m-2)\,C\right]P_s + O(P_s^2) \qquad (298)$$

Substituting (298) into (63) yields (64). ∎

APPENDIX D

Proof of Corollary 10.

For $x = (m-2)\,P_s$ and $n = C$, (290) yields

$$[1 + (m-2)\,P_s]^C$$
$$\approx 1 + C(m-2)\,P_s + \frac{C(C-1)}{2}[(m-2)\,P_s]^2$$
$$+ \frac{C(C-1)(C-2)}{6}[(m-2)\,P_s]^3$$
$$+ \frac{C(C-1)(C-2)(C-3)}{24}[(m-2)\,P_s]^4 \ . \quad (299)$$

Also, for $x = -P_s$ and $n = (m-2)\,C$, (290) yields

$$(1 - P_s)^{(m-2)\,C}$$
$$\approx 1 - (m-2)\,C P_s + \frac{(m-2)\,C[(m-2)\,C - 1]}{2}P_s^2$$
$$- \frac{(m-2)\,C[(m-2)\,C - 1][(m-2)\,C - 2]}{6}P_s^3$$
$$+ \frac{(m-2)\,C[(m-2)\,C - 1][(m-2)\,C - 2][(m-2)\,C - 3]}{24}P_s^4 \ .$$
$$(300)$$

From (126), (299), and (300), it follows that

$$q_1^C \approx 1 - \frac{C(m-1)(m-2)}{2}P_s^2 + \frac{C(m-1)(m-2)(m-3)}{3}P_s^3$$
$$+ \frac{C(m-1)(m-2)}{8}[C(m-1)(m-2) - 2(m^2 - 5m + 7)]P_s^4 \ .$$
$$(301)$$

By setting $C = 1$, (301) yields

$$q_1 \approx 1 - \frac{(m-1)(m-2)}{2}P_s^2 + \frac{(m-1)(m-2)(m-3)}{3}P_s^3$$
$$- \frac{(m-1)(m-2)(m-3)(m-4)}{8}P_s^4 \ . \quad (302)$$

Substituting (301) and (302) into (56) yields

$$P_{\text{UF in } S_1|2} \approx 1$$
$$- \frac{12 - 8(m-3)P_s - 3[C(m-1)(m-2) - 2(m^2 - 5m + 7)]P_s^2}{12 - 8(m-3)P_s + 3(m-3)(m-4)P_s^2}$$
$$= \frac{3(C-1)(m-1)(m-2)P_s^2}{12 - 8(m-3)P_s + 3(m-3)(m-4)P_s^2}$$
$$\approx \frac{(C-1)(m-1)(m-2)}{4}P_s^2 \ . \quad (303)$$

∎

APPENDIX E

Proof of Corollaries 12 and 13.

From (126) and (147) it follows that

$$q_1 = (1 + x)\,q_2 \ , \quad (304)$$

where

$$x \triangleq (m-2)P_s \ . \quad (305)$$

Then it holds that

$$\frac{q_2}{C}\,\frac{q_1^C - q_2^C}{q_1 - q_2} \overset{(304)}{=} \frac{q_2}{C}\,\frac{[(1+x)^C - 1]q_2^C}{x\,q_2} = \frac{[(1+x)^C - 1]}{C\,x}\,q_2^C$$
$$\overset{(148)(305)}{\approx} \frac{[(1 + Cx + \frac{C(C-1)}{2}x^2 - 1]}{C\,x}(1 - x)^C$$
$$\approx \left[1 + \frac{C-1}{2}x\right](1 - C\,x)$$
$$= 1 - \frac{C+1}{2}x + O(x^2)$$
$$= 1 - \frac{C+1}{2}(m-2)\,P_s + O(P_s^2) \quad (306)$$

and

$$q_2^2\,\frac{q_1^C - C\,q_1\,q_2^{C-1} + (C-1)q_2^C}{(q_1 - q_2)^2}$$
$$\overset{(304)}{=} q_2^2\,\frac{[(1+x)^C - C\,(1+x) + (C-1)]q_2^C}{(x\,q_2)^2}$$
$$\approx \frac{[(1 + Cx + \frac{C(C-1)}{2}x^2 + \frac{C(C-1)(C-2)}{6}x^3 - Cx - 1)]}{x^2}q_2^C$$
$$\overset{(148)(305)}{\approx} \left[\frac{C(C-1)}{2} + \frac{C(C-1)(C-2)}{6}x\right](1 - x)^C$$
$$\approx \frac{C(C-1)}{2}\left[1 + \frac{C-2}{3}x\right](1 - C\,x)$$
$$= \frac{C(C-1)}{2} - \frac{C\,(C-1)\,(C+1)}{3}x + O(x^2)$$
$$\overset{(305)}{=} \frac{C(C-1)}{2} - \frac{C\,(C-1)\,(C+1)}{3}(m-2)\,P_s + O(P_s^2)$$
$$(307)$$

Substituting (306) into (168) yields the first part of (179). Given that $P_{\text{UF,2}} = P_{\text{UF}|2}\,P_{\text{DF,2}} \le P_{\text{DF,2}}$, this part is valid when $P_{\text{UF,2}} = A\,P_s \le P_{\text{DF,2}}$ or, equivalently, $P_s \le P_{\text{DF,2}}/A$, which by virtue of (162), (177), and (182) implies that $P_s \le P_{s*}^{(2)}$. For $P_s \gg P_{s*}^{(2)}$, $P_{\text{UF,2}} \approx P_{\text{DF,2}}$, which is the second part of (179).

∎

APPENDIX F

Proof of Proposition 1.

The probability $P_{\text{UF,3}}$ of data loss due to unrecoverable failures, given that three device failures have occurred, is obtained by unconditioning (208) on i and j, and using (201) as follows:

$$P_{\text{UF,3}} = \sum_{j=1}^{C}\sum_{i=j+1}^{C} P_{\text{UF}|3}(j,i)\,P_{j,i}$$
$$\approx \sum_{j=1}^{C}\sum_{i=j+1}^{C}\left(1 - q_1^{j-1}\,q_2^{i-j}\,p_3^{C-i+1}\right)\frac{(m-1)(m-2)}{C^2}\left(\frac{\lambda}{\mu}\right)^2 f_R \ .$$
$$(308)$$

It holds that

$$\sum_{j=1}^{C}\sum_{i=j+1}^{C} 1 = \sum_{j=1}^{C}(C-j) = \frac{C\,(C-1)}{2} \ , \quad (309)$$

and

$$\sum_{j=1}^{C} \sum_{i=j+1}^{C} q_1^{j-1} \, q_2^{i-j} \, p_3^{C-i+1}$$

$$= \sum_{j=1}^{C} q_1^{j-1} \, q_2 \, p_3^{C-j} \sum_{i=j+1}^{C} q_2^{i-j-1} \, p_3^{j-i+1}$$

$$= \sum_{j=1}^{C} q_1^{j-1} \, q_2 \, p_3^{C-j} \sum_{k=0}^{C-j} \left(\frac{q_2}{p_3}\right)^k$$

$$= \sum_{j=1}^{C} q_1^{j-1} \, q_2 \, p_3^{C-j} \frac{1 - \left(\frac{q_2}{p_3}\right)^{C-j}}{1 - \frac{q_2}{p_3}}$$

$$= \frac{q_2 \, p_3}{p_3 - q_2} \sum_{j=1}^{C} q_1^{j-1} \left(p_3^{C-j} - q_2^{C-j}\right)$$

$$= \frac{q_2 \, p_3}{p_3 - q_2} \left(\frac{q_1^C - p_3^C}{q_1 - p_3} - \frac{q_1^C - q_2^C}{q_1 - q_2}\right) . \qquad (310)$$

Substituting (309) and (310) into (308) yields (209). ∎

APPENDIX G

Proof of Corollary 15.

For small values of P_s, and from (127) , (148), and (207), it follows that

$$q_1^{j-1} \approx 1 - \frac{(j-1)(m-1)(m-2)}{2} P_s^2 , \qquad (311)$$

$$q_2^{i-j} \approx 1 - (i-j)(m-2) P_s , \qquad (312)$$

$$p_3^{C-i+1} \approx 1 - (C-i+1)(m-3) P_s . \qquad (313)$$

Consequently,

$$q_1^{j-1} \, q_2^{i-j} \, p_3^{C-i+1}$$
$$\approx 1 - [(i-j)(m-2) + (C-i+1)(m-3)]P_s + O(P_s^2)$$
$$\approx 1 - [(C+1)(m-3) - (m-2)j + i] P_s . \qquad (314)$$

Therefore,

$$\sum_{j=1}^{C} \sum_{i=j+1}^{C} \left(1 - q_1^{j-1} \, q_2^{i-j} \, p_3^{C-i+1}\right)$$

$$\approx \sum_{j=1}^{C} \sum_{i=j+1}^{C} [(C+1)(m-3) - (m-2)j + i]P_s$$

$$\approx \sum_{j=1}^{C} (C-j)\,[(C+1)(m-3) - (m-2)j]P_s + \sum_{j=1}^{C} \sum_{i=j+1}^{C} i \, P_s$$

$$\approx [(C+1)(m-3)]P_s \sum_{j=1}^{C} (C-j)$$

$$- (m-2) \sum_{j=1}^{C} [(C-j)\,j] \, P_s + \sum_{j=1}^{C} \sum_{k=0}^{C-j-1} (k+j+1) \, P_s$$

$$\approx [(C+1)(m-3)]P_s \frac{(C-1)\,C}{2}$$

$$- (m-2) \sum_{j=1}^{C} [(C-j)\,j] \, P_s + \sum_{j=1}^{C} \sum_{k=0}^{C-j-1} (k+j+1) \, P_s$$

$$\approx \frac{(C-1)\,C\,(C+1)}{2} (m-3) \, P_s - (m-2) \sum_{j=1}^{C} [(C-j)\,j] \, P_s$$

$$+ \sum_{j=1}^{C} \left[(C-j)(j+1) + \frac{(C-j-1)(C-j)}{2} \right] P_s$$

$$\approx \frac{(C-1)\,C\,(C+1)}{2} (m-3) \, P_s - (m-3) \sum_{j=1}^{C} [(C-j)\,j] \, P_s$$

$$+ \frac{1}{2} \sum_{j=1}^{C} (C-j) \, P_s + \frac{1}{2} \sum_{j=1}^{C} (C-j)^2 \, P_s$$

$$\approx \frac{(C-1)\,C\,(C+1)}{2} (m-3) \, P_s$$

$$- (m-3) \frac{(C-1)\,C\,(C+1)}{6} P_s$$

$$+ \frac{1}{2} \frac{(C-1)\,C}{2} P_s + \frac{1}{2} \frac{(C-1)\,C\,(2\,C-1)}{6} P_s$$

$$\approx \frac{(C-1)\,C\,(C+1)}{3} (m-3) \, P_s$$

$$+ \frac{1}{2} \frac{(C-1)\,C}{2} P_s + \frac{1}{2} \frac{(C-1)\,C\,(2\,C-1)}{6} P_s$$

$$\approx \frac{(C-1)\,C}{6} \left[2\,(C+1)(m-3) + \frac{3}{2} + \frac{2\,C-1}{2}\right] P_s$$

$$\approx \frac{(C-1)\,C\,(C+1)}{6} (2\,m-5) \, P_s \qquad (315)$$

Substituting (315) into (308) yields the first part of (210). Given that $P_{\mathrm{UF},3} = P_{\mathrm{UF}|3} \, P_{\mathrm{DF}_3} \le P_{\mathrm{DF}_3}$, this part is valid when $P_{\mathrm{UF},3} = B \, P_s \le P_{\mathrm{DF}_3}$ or, equivalently, $P_s \le P_{\mathrm{DF}_3}/B$, which by virtue of (205) and (212) implies that $P_s \le 3/(C\,(2m-5))$, which is of the same order as $P_{s*}^{(2)}$. For $P_s \gg P_{s*}^{(2)}$, $P_{\mathrm{UF},3} \approx P_{\mathrm{DF}_3}$, which is the second part of (210). ∎

REFERENCES

[1] I. Iliadis, "Data loss in RAID-5 storage systems with latent errors," in Proceedings of the 12th International Conference on Communication Theory, Reliability, and Quality of Service (CTRQ), Mar. 2019, pp. 1–9.

[2] D. A. Patterson, G. Gibson, and R. H. Katz, "A case for redundant arrays of inexpensive disks (RAID)," in Proceedings of the ACM SIGMOD International Conference on Management of Data, Jun. 1988, pp. 109–116.

[3] P. M. Chen, E. K. Lee, G. A. Gibson, R. H. Katz, and D. A. Patterson, "RAID: High-performance, reliable secondary storage," ACM Comput. Surv., vol. 26, no. 2, Jun. 1994, pp. 145–185.

[4] V. Venkatesan, I. Iliadis, C. Fragouli, and R. Urbanke, "Reliability of clustered vs. declustered replica placement in data storage systems," in Proceedings of the 19th Annual IEEE/ACM International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems (MASCOTS), Jul. 2011, pp. 307–317.

[5] I. Iliadis, D. Sotnikov, P. Ta-Shma, and V. Venkatesan, "Reliability of geo-replicated cloud storage systems," in Proceedings of the 2014 IEEE 20th Pacific Rim International Symposium on Dependable Computing (PRDC), Nov. 2014, pp. 169–179.

[6] M. Malhotra and K. S. Trivedi, "Reliability analysis of redundant arrays of inexpensive disks," J. Parallel Distrib. Comput., vol. 17, Jan. 1993, pp. 146–151.

[7] A. Thomasian and M. Blaum, "Higher reliability redundant disk arrays: Organization, operation, and coding," ACM Trans. Storage, vol. 5, no. 3, Nov. 2009, pp. 1–59.

[8] I. Iliadis, R. Haas, X.-Y. Hu, and E. Eleftheriou, "Disk scrubbing versus intradisk redundancy for RAID storage systems," ACM Trans. Storage, vol. 7, no. 2, Jul. 2011, pp. 1–42.

[9] V. Venkatesan, I. Iliadis, and R. Haas, "Reliability of data storage systems under network rebuild bandwidth constraints," in Proceedings of the 20th Annual IEEE International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems (MASCOTS), Aug. 2012, pp. 189–197.

[10] J.-F. Pâris, T. J. E. Schwarz, A. Amer, and D. D. E. Long, "Highly reliable two-dimensional RAID arrays for archival storage," in Proceedings of the 31st IEEE International Performance Computing and Communications Conference (IPCCC), Dec. 2012, pp. 324–331.

[11] I. Iliadis and V. Venkatesan, "Most probable paths to data loss: An efficient method for reliability evaluation of data storage systems," Int'l J. Adv. Syst. Measur., vol. 8, no. 3&4, Dec. 2015, pp. 178–200.

[12] ——, "Expected annual fraction of data loss as a metric for data storage reliability," in Proceedings of the 22nd Annual IEEE International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems (MASCOTS), Sep. 2014, pp. 375–384.

[13] ——, "Reliability evaluation of erasure coded systems," Int'l J. Adv. Telecommun., vol. 10, no. 3&4, Dec. 2017, pp. 118–144.

[14] I. Iliadis, "Reliability evaluation of erasure coded systems under rebuild bandwidth constraints," Int'l J. Adv. Networks and Services, vol. 11, no. 3&4, Dec. 2018, pp. 113–142.

[15] Amazon Simple Storage Service. [Online]. Available: http://aws.amazon.com/s3/ [retrieved: January 2019]

[16] D. Borthakur et al., "Apache Hadoop goes realtime at Facebook," in Proceedings of the ACM SIGMOD International Conference on Management of Data, Jun. 2011, pp. 1071–1080.

[17] R. J. Chansler, "Data availability and durability with the Hadoop Distributed File System," ;login: The USENIX Association Newsletter, vol. 37, no. 1, 2013, pp. 16–22.

[18] K. Shvachko, H. Kuang, S. Radia, and R. Chansler, "The Hadoop Distributed File System," in Proceedings of the 26th IEEE Symposium on Mass Storage Systems and Technologies (MSST), May 2010, pp. 1–10.

[19] Hitachi Global Storage Technologies, Hitachi Disk Drive Product Datasheets. [Online]. Available: http://www.hitachigst.com/ [retrieved: January 2019]

[20] E. Pinheiro, W.-D. Weber, and L. A. Barroso, "Failure trends in a large disk drive population," in Proceedings of the 5th USENIX Conference on File and Storage Technologies (FAST), Feb. 2007, pp. 17–28.

[21] A. Dholakia, E. Eleftheriou, X.-Y. Hu, I. Iliadis, J. Menon, and K. Rao, "A new intra-disk redundancy scheme for high-reliability RAID storage systems in the presence of unrecoverable errors," ACM Trans. Storage, vol. 4, no. 1, May 2008, pp. 1–42.

[22] I. Iliadis, "Reliability modeling of RAID storage systems with latent errors," in Proceedings of the 17th Annual IEEE/ACM International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems (MASCOTS), Sep. 2009, pp. 111–122.

[23] V. Venkatesan and I. Iliadis, "Effect of latent errors on the reliability of data storage systems," in Proceedings of the 21th Annual IEEE International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems (MASCOTS), Aug. 2013, pp. 293–297.

[24] I. Iliadis and V. Venkatesan, "Rebuttal to 'Beyond MTTDL: A closed-form RAID-6 reliability equation'," ACM Trans. Storage, vol. 11, no. 2, Mar. 2015, pp. 1–10.

[25] V. Venkatesan and I. Iliadis, "A general reliability model for data storage systems," in Proceedings of the 9th International Conference on Quantitative Evaluation of Systems (QEST), Sep. 2012, pp. 209–219.

[26] I. Iliadis and X.-Y. Hu, "Reliability assurance of RAID storage systems for a wide range of latent sector errors," in Proceedings of the 2008 IEEE International Conference on Networking, Architecture, and Storage (NAS), Jun. 2008, pp. 10–19.

[27] V. Venkatesan and I. Iliadis, "Effect of codeword placement on the reliability of erasure coded data storage systems," in Proceedings of the 10th International Conference on Quantitative Evaluation of Systems (QEST), Sep. 2013, pp. 241–257.

[28] I. Iliadis and V. Venkatesan, "An efficient method for reliability evaluation of data storage systems," in Proceedings of the 8th International Conference on Communication Theory, Reliability, and Quality of Service (CTRQ), Apr. 2015, pp. 6–12.

[29] V. Venkatesan and I. Iliadis, "Effect of codeword placement on the reliability of erasure coded data storage systems," IBM Research Report, RZ 3827, Aug. 2012.

Media Comparison for Instruction-based AR Usage in Collaborative Assembly

Lea M. Daling, Anas Abdelrazeq, Frank Hees, and Ingrid Isenhardt

Chair of Information Management in Mechanical Engineering (IMA)

RWTH Aachen University

Aachen, Germany

lea.daling@ima-ifu.rwth-aachen.de, anas.abdelrazeq@ima.rwth-aachen.de, frank.hees@ima.rwth-aachen.de,
ingrid.isenhardt@ima.rwth-aachen.de

Abstract— **Complex work processes are increasingly performed as a cooperation between humans and robots. In this context, assistive technologies play an important role. The implementation of Augmented Reality as instructional tool offers new possibilities to support workplace-oriented learning processes. This paper studies the suitability and usability of different AR-based media in collaborative assembly between human and robot. For this purpose, three different media have been compared in an experimental within-subject design with regard to the usability of the respective hardware and software. The findings gained from the mixed-method approach show that traditional media are considered easier to use, but that the fundamental potential of Augmented Reality application is clearly recognized.**

Keywords - Augmented Reality; Human-Robot-Interaction; Usability Evaluation; On-the-Job Training; Assistive Technology.

I. INTRODUCTION

New assistance technologies are finding their way into formerly manually and analogously designed areas of the digitized industrial working world. Hereby, employees can be directly supported in the work process [1] and are enabled to cope with new and complex requirements of an increasingly individualized and highly flexible production [2]. The use of Augmented Reality (AR) as an instructional assistance tool is widely expected to be a success factor for digital training programs [3]. It allows employees to be guided through assembly processes step by step, and to train them flexibly for new use cases.

Well-designed assistance systems become particularly important when employees have to be trained for complex or novel work processes. In the course of increased interaction between man and machine, scenarios in which both actors simultaneously work together on a task become more widespread [4]. Therefore, this paper focuses on the design and evaluation of AR-based assistance systems in assembly processes that are neither completely manual nor fully automated and take place in cooperation between human and robot [5]. The presented AR application is designed for the instruction of a collaborative assembly process between human and robot.

Since a high degree of usability can be seen as a prerequisite for further performance measures such as time effectiveness and reducing the error rate, the aim of our current research is to test the usability of an AR-based assistance system tested on different media [6]. This paper represents a detailed elaboration of a usability evaluation for the use of AR assistance systems in human-robot collaboration [1]. In Section II a brief introduction of the basic functions and application possibilities of AR in the manufacturing context are given. The use case of AR as an on-the-job instructional tool in a collaborative assembly cell is presented in Section III. In Section IV, an overview of relevant usability criteria is presented. Furthermore, we present our empirical approach to measure usability of an AR application using different instructional media. Finally, Section V gives an overview of the results and an outlook on further research.

II. THEORETICAL BACKGROUND

The following paragraph gives a brief introduction of the basic functions and application possibilities of AR in manufacturing and the use of AR as an on-the-job instructional tool in a collaborative assembly cell.

A. Instructional AR in the Manufacturing Context

An AR system adds virtual objects to the real world, in a way that both virtual and real components homogeneously appear in the user perception. An AR system "combines real and virtual objects in a real environment; runs interactively and in real time and registers (aligns) real and virtual objects with each other" [7]. In other words, AR systems overlay computer-generated objects onto a real world setting, in real time [8].

Within the last 10-15 years, AR systems greatly improved and have shown an ability to create solutions to various problems [9]. Since then, more and more AR tools are developed and applied in the field of industry. The main use of AR in an industrial context is currently related to maintenance, manufacturing, and assembly related tasks [10]. Using AR, innovative and effective methods can be developed to meet important requirements in simulation, assistance and improvement of manufacturing processes. Volvo, for example, is utilizing the Microsoft HoloLens to enable production line workers to digitally view assembly instructions in real-time while working to put together parts of the vehicle [11]. By adding real-time information to a real (working) environment, AR-based systems can minimize the need for improvement iterations, re-works and modifications by 'getting it done the right way' on the first try.

As a result, the possibility of "learning on demand" in on-the-job training sessions arises. To date, there are several approaches that combine learning measures at the workplace with the benefits of new technologies [3]. These on-the-job learning approaches connect theoretical knowledge with practical application [18]. Furthermore, they provide tailor-made learning processes and can be used independent of

time and learning pace [3]. The use of AR in training processes promises many positive effects, such as constant access to information, lower error rate, improved motivation and a synchronization between training and performance [13]. For instance, a comparison between paper instructions and AR instructions on a Head Mounted Display (HMD) showed that, although the use of AR in the assembly process gives little "time-advantages", it significantly reduces the assembly errors [12].

Nevertheless, AR systems still face a couple of challenges, preventing a direct implementation of AR solutions in real world problems. Technical developments are often limited to the capabilities of the specific medium instead of analyzing the requirements of the respective task or work process. The focus is therefore more on the technology to be implemented than on the user or the requirements of the task. Many studies already focus on objective key indicators (e.g., time, error rate, accuracy) as dependent variables, even though basic usability factors such as acceptance, perceived usefulness or enjoyment as well as technical questions (e.g., display and tracking technology, calibration techniques, interfaces to the operating devices [13]) should be of central importance and precede the implementation process [14].

Even with those challenges conquered, other questions still arise. Like whether or not the implementation of such systems would lead to other problems affecting the overall performance. An over-reliance on the AR generated signals and indications can have negative implications on the performance of the user, for example by disrupting the attention or focusing it all in one direction, leading it away from the surrounding context [17]. Further research and evaluation of the technology is therefore necessary to solve existing problems and expand the spectrum of applications.

B.   AR as an interface for collaborative assembly

The current developments in connection with the increasingly networked and individualized Industry not only impose high demands on interconnected technical systems. More complex, dynamic and individualized production processes are also changing the way work as such is organized. Increased interaction between humans and robots is considered to be a future scenario, in which tasks are accomplished together while working on a product or a component at the same time. Particularly in manufacturing, tasks change from manual work to collaborative work processes between humans and robots or machines. While cooperation means that both interacting partners can have tasks in the (common) workspace, but do not work on the same product or component at the same time, collaboration describes the simultaneous execution of a common task on the same product or component (see **Figure 1**) [4].

On the one hand, this changes the requirements placed on the employees who interact with these machines [2]. On the other hand, it changes the requirements placed on assistance systems, which are intended to provide the best possible support for the fulfilment of the respective task. As already mentioned, AR is potentially suitable as an interface between humans and robots, in order to guide through new tasks and
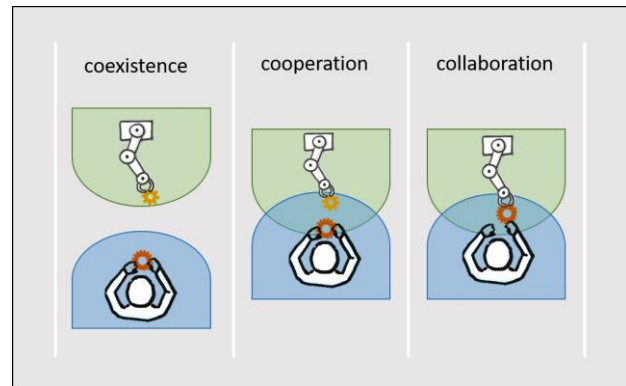
Figure 1.   Levels of Human-Robot Interaction [4]

procedures in on-the-job trainings. In addition, when used as an assistance system, it can also create transparency about the current status or work steps of the robot.

AR as an assistant system for collaborative tasks can be used, to display assembly process information, robot motion and workspace visualization, visual alerts, and production data amongst others [15]. Makris and colleagues, for example, showed that the AR application was able to minimize the time required for operators to access the necessary information and also increased the operator's acceptance to work with industrial robots without safety fences. Michalos and colleagues used an AR based application that ran on an Android tablet to support human-robot interactive cooperation [16]. Although they achieve positive results, they point out that the AR application should be tested on different media (e.g., head-mounted) and that overall, more focus should be directed towards researching factors such as ergonomics and handling of such media, as well as the design of the AR application itself (e.g., layout of visual aids).

This leads to the assumption that, before the effectiveness of AR and its appropriateness for the use case of human robot collaboration for assembly tasks can be tested, there is still a need for research regarding the usability of AR systems and different media. In addition to that, it is important to gain a deeper understanding on how different aspects of the AR system are perceived and evaluated by the user. In this respect, it is important to first analyze the task or requirements of the work process. The technical development should be iterative and adapted to the needs of the users. When evaluating the AR assistance system, it is advisable to distinguish between hardware (e.g., data glasses vs. tablet) and software (the AR application used) [1]. Therefore, the present qualitative pre-study aims at deriving basic implications on the usability of the developed AR application by not only providing feedback on the AR-capable hardware, but also on the AR application software itself. Throughout the next section, the usability aspects to be considered are explained before the use case and the analysis of the task are presented.

C.   Usability Aspects

Since 1997, DIN EN ISO 9241 has been an international series of standards that defines usability as the extent to

which a technical system can be used by certain users in a certain usage context in order to achieve certain goals effectively, efficiently and satisfactorily [6]. Sarodnick and Brau emphasize that usability particularly considers the fit of system, task and user, while taking into account the quality of goal achievement perceived by the user [6]. For this reason, it is essential to involve potential users in the evaluation process at an early stage.

A survey of Gabbard and colleagues [20] showed that in a total of 1104 articles on augmented reality, only 38 (~3%) addressed some aspect of human computer interaction, and only 21 (~2%) described a formal user-based study. Since, as mentioned, the involvement of users in the evaluation process is crucial for the successful development of a product, a user-centered mixed-method approach will be presented in the following.

A widely used *inductive* approach, characterized by the analysis of early versions and prototypes, is the so-called thinking aloud method [6]. Here, test subjects are encouraged to express their cognitions verbally during the test. The advantage of this approach is the explorative acquisition of qualitative data to receive feedback on design and improvement. However, it should be critically noted that the combined load of task processing and thinking aloud reduces the processing speed. Therefore, this method should not be used in conjunction with a performance measurement. Furthermore, these approaches are barely standardized.

Deductive methods, on the other hand, capture the user's perspective on an already developed system. At this point, however, changes and corrections of a system are often time- and cost consuming. Established evaluation concepts (e.g., IsoMetrics; Isonorm [22]), often make use of the classical questionnaire methodology, which ensures the fulfilment of the quality criteria (validity, reliability, objectivity) to a large extend. The aim of this paper is to combine the advantages of both methods in order to generate feedback on the usability of the AR application based on empirical user surveys - extended by open questions. Furthermore, the thinking aloud method was used to verbalize and record the impressions, reactions and cognitions of the participants during the work process. Before the composition of these approaches is presented in Section IV, the respective use case is presented in the following section.

III. USE CASE AND REQUIREMENTS

The use case in which AR is utilized to enable on-the-job training consists of a collaborative assembly cell equipped with a robot. It represents a common scenario in Industry 4.0, where the digitalization of production is continuously increasing. It also poses special challenges to the interface design of the instructional tool, due to the interaction between humans and robots.

During the assembly process, workers are collaborating with a robot arm (UR-5) to assemble a small gear drive. In total, three plates with gear wheels are assembled. The worker performs five steps, while the robot performs a total of four steps. Once the worker has familiarized himself with the cell, he/she is instructed to position a base plate and rear plate into a holder. The robot then inserts four hexagon

socket screws and positions the back plate onto the base plate, while the worker assembles two sets of gear wheels. In the final step, the gear wheels are mounted on the pre-assembled base plate presented by the robot.

Previous studies have shown that an efficient and user-friendly introduction training can contribute significantly to increasing the acceptance of the human-robot interaction [21]. During these studies, the assembly task was guided by a fixed touchscreen with 2D images and 3D animations. Thereby, participants repeatedly had to look up the work steps on the fixed screen.

AR offers the benefit of displaying information and work steps, e.g., 3D overlays, directly within the workspace or the tool required for the respective assembly step. The instructions for the AR application were developed based on the existing work steps and supplemented by virtual objects with real-time animations. During the design of the work instructions, special attention was paid to the fact that the instructions must be comprehensible for laypersons and inexperienced employees. Based on fundamental usability heuristics (e.g., visibility of the system status, consistency and aesthetics) [19] and iterative testing within the interdisciplinary development team, we defined the following requirements for the development the AR-Application:

1. The application is designed to provide non-experts with an **easy and intuitive on-the-job training process** without moving back to the screen.

2. The application should function on **head-mounted** and **handheld AR devices.**

3. The application should be able to **recognize the working space.**

4. The application should illustrate different **step cues** (text, 2D images and 3D animations) for the user.

5. The application should enable a predominantly **hands-free instruction process**. Thus, the worker should be able to either **navigate with touch or voice** control.

6. To ensure a good workflow between human and robot, the AR application should be able to **communicate with the robot** and be aware of its status.

In order to meet these requirements, an AR application was developed that is deployable on both Microsoft HoloLens and Android-based tablets fixed with a tablet arm (see **Figure 2**).

Figure 2. Using HoloLens and Android Tablet as Instructional Media

Unity 3D game engine was adapted for the software development process. The proposed AR application consists of several modules (e.g., game objects) that are organized within a Unity scene (see **Figure 3**). Different components, including the interface layout, are realized within these modules.

In order to provide orientation at the workplace and to locate the devices optimally within the assembly cell, special attention was paid during development to implement *Localization and Tracking*. For this purpose, one marker was placed in the cell for the HoloLens, whereas the tablet uses two different markers for the different work areas in the cell due to its lower tracking capability. If the tracking has been completely lost, the application asks the user to reposition the device to the point where the marker is seen. In addition, guidance arrows are used to indicate the location of the next step. In order to enable dynamic and on demand changes, as part of the *scenario manager*, a configuration ".json" file was developed to configure the steps specifying their text instructions, 2D images and 3D animations, as well as information about the robot status. The *interactive user interface* allows the user to navigate through the steps by either by touching (i.e., clicking) the tablet or by voice control of the HoloLens. *Robot Communication* is ensured based on Robot Operating System (ROS) to initiate a step or to wait for the robot until it finished its step.
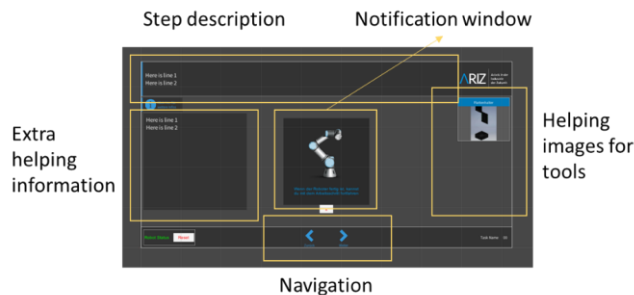

Figure 3.   Layout of the AR Application

IV.   METHOD

The aim of the present usability evaluation is to collect feedback on an AR application prototype that is tested on different media. The chosen mixed-method approach combines inductive qualitative methods with deductive, quantitatively oriented approaches of data acquisition. In 1993, Nielsen stated that a number of 5-6 test subjects were sufficient to detect significant problems [23]. Since not only the AR application but also the usability of the three media used is to be evaluated, we aimed at a minimum N of 15 persons. According to Faulkner, at least 90% to 97% of all known usability problems can be detected with a number of 15 people [24]. Therefore, we decided on a within-subject design in which every test person performs tests on every medium. The study design, the description of the sample and the used questionnaires will be presented in the following section.

A.   Study Design and Procedure

In addition to the evaluation of the AR application, the usability of the respective instructional media should also be evaluated. Thus, we have set up a within-subject test design, where the participants have to perform three rounds on the assembly cell (1. Tablet (AR); 2. HoloLens (AR); 3. Touchscreen (non-AR)). Each round was instructed by different instructional media: The AR application is used by two media (the tablet and the HoloLens), so that the evaluation of the AR application can be carried out independently of the medium used. In order to compare these media with previously used media, the touchscreen is also included in the testing. It uses text- and animation-based instructions but is not AR-capable and therefore limited to the dimensionality of the screen. In order to control for repetition and learning effects [25] as far as possible, the order of the instruction media was randomized.

Each participant completed a pre-test questionnaire at the beginning in a paper-pencil format. They were then asked to familiarize themselves with the workstation of the assembly cell. Depending on the randomized condition, the first assembly was instructed by either the tablet, the HoloLens or the touchscreen. The participants had the opportunity to ask the test supervisor for help at any time, but were encouraged to carry out the assembly themselves. After each assembly process, which was completed as soon as the fully assembled gear drive was placed in a box by the robot, there was a post-test questionnaire referring to the medium used. During all three sessions, the subjects were encouraged to express their thoughts aloud. The statements were recorded with a voice recorder. After the third assembly had been completed, participants were asked to fill out the third part of the questionnaire referring to the AR-application itself. The study took about 60 minutes to complete.

B.   Participants

A total of eight men and seven women took part in the study (N = 15). The mean age of the study participants was 25 years (MW = 25.07, range = 20 - 32). The sample consisted of eleven students and four working persons. Seven participants indicated to have high school graduation and/or the general university entrance qualification as the highest education degree, the remaining eight already have an academic degree (nBachelor = 4, nMaster = 3, nPhD = 1). Twelve participants have never worked with a robot; the other three have rarely worked with a robot. Only one person had already participated in a study on the collaborative assembly cell.

TABLE I.   SCALES – INSTRUCTIONAL MEDIA

Scales	Sources	Number of Items	Final number and Cronbachs α	Example Items
Task-Load	NASA Task Load Index [29]	6 Items	6 Items Cronbachs α = .68 - .83*	"How much mental and perceptual activity was required? Was the task easy or demanding, simple or complex?"
Perceived Usefulness	Technology Acceptance Model (TAM 3) – Perceived Usefulness [30]	4 Items	4 Items Cronbachs α = .91 - .93	"Using the instruction medium would improve my work performance."
Media Self-Efficacy	TAM 3 – Computer Self-Efficacy [30]	4 Items	2 Items Cronbachs α = .85 - .95	"I would be able to use the instructional medium to do my work if no one were present to tell me what to do."
Perceived Enjoyment	TAM 3 – Perceived Enjoyment [30] Key Components of User Experience (meCue2.0; [28]	3 Items (TAM 3) 3 Items (meCue2.0)	3 Items (TAM 3) 3 Items (meCue2.0) Cronbachs α = .66 - .92	"I would enjoy using the instructional medium.""The instructional medium frustrates me."
Perceived Ease of Use	TAM 3 – Perceived Ease of Use [30] IsoMetrics [22]	4 Items (TAM 3) 2 Items (IsoMetrics)	4 Items (TAM 3) 1 Item (IsoMetrics) Cronbachs α = .56 - .89	"I think the handling of the instructional medium would be clear and understandable for me.""The operating options of the instructional medium support an optimal use of the application."
Open Questions	• What did you particularly like about the instruction medium you used? • What would have to be changed on the instruction medium to make the assembly process even easier? • Please create a ranking of the instruction media, 1 being your strongest preference, 2 being the second choice, etc. Justify your decision.			

* Cronbachs α has been evaluated for three different media and therefore is presented as a range.

C. Questionnaires

In the following paragraph, the pre- and post-test questionnaires for both "Instructional Media" and the "AR application" are presented. An overview of all scales with example items can be seen in **TABLE I**.

1) Pre-Test.

In addition to the demographic data already reported, the participants were asked about their affinity for technology with five items (e.g., "My enthusiasm for technology is...") on a six-level scale ranging from "very low" to "very high". To complete the data on the participants, we also asked which media (e.g., laptop, smartphone or tablets) are available to them, how often they use them and how easy it is to use the respective medium. In addition, we used the "locus of control for technology" questionnaire (KUT) to assess general control beliefs while dealing with technology [26]. With its eight items (e.g., "Most of the technological problems that I have to face can be solved by myself") on a six-level scale ranging from "not true at all" to "absolutely true" the German questionnaire has a reliability of α = 0.89 [26].

In order to measure the participant's mood before and after the collaborative work process, we decided to use the Affect Grid [27]. The Affect Grid has been designed as a rapid means of evaluating affects in the dimensions of pleasant-unpleasant and arousal-sleepiness. The scale shows reasonable reliability, convergent validity and discriminant validity in studies in which subjects used the Affect Grid to describe their current mood. Since the Affect Grid is particularly suitable for repeated use, the mood was measured after each run with the various media.

2) Post-Test – Instructional Media.

The assessment of the usability of the instructional media used is carried out separately from the evaluation of the AR application. Thus, it is possible to separate the findings on software and hardware more clearly. Based on existing usability literature [6]][19][22][28], we decided to select relevant and quantifiable criteria for the task with regard to their face validity in order to determine the suitability of the chosen instructional media.: *a) task load, b) perceived usefulness, c) media self-efficacy, d) perceived enjoyment,* and *e) perceived ease of use.*

a) Task load. The task load was measured by the "NASA Task Load Index (NASA TLX)". It measures subjectively experienced demand using a multidimensional scale that differentiates, for example, between physical and mental strain [29]. The German short version contains six dimensions, namely; mental, physical and temporal demands, as well as performance, effort and frustration. The original scale has 20 gradations from "very low" to "very high". Adapted to the German version, we used a 10-step scale with the poles "little" and "much". Criteria on reliability have been satisfactorily reviewed (Cronbachs α = .68 - .83).

b) Perceived usefulness. The factor perceived usefulness arises from the widespread and empirically well-founded "Technology Acceptance Model (TAM)" [30], which has

been incorporated into the development of the usability catalogue. The TAM, currently in its third version, aims at predicting the usage behavior and acceptance of information technologies. To represent the construct, we used four items on a scale from one (strongly disagree) to seven (strongly agree) and adapted them to our application (e.g., "using the instruction medium would improve my work performance"). Cronbach's alpha showed a satisfactory value of $\alpha = .91$ -.93.

c) Media self-efficacy. Four items from TAM 3's original "Computer Self-Efficacy" scale were used and subsequently adapted (e.g., "I would be able to use the instructional medium to do my work if no one were present to tell me what to do"). Since two of these items - presumably due to a misleading formulation – showed a high standard deviation, they were excluded from further analysis. The remaining two items reached a Cronbach's alpha of $\alpha = .85$ - .95.

d) Perceived enjoyment. This construct is composed of three adapted items from TAM 3 (perceived enjoyment; e.g., "I would enjoy using the instructional medium.") and three other items from the "Modular Evaluation of Key Components of User Experience"(meCue2.0; e.g., "The instructional medium frustrates me."). This questionnaire is based on the analytical "Components of User Experience" model by Thüring and Mahlke [28]. This model distinguishes between the perception of task-related and non-task-related product qualities and includes user emotions as an essential and mediative factor of certain usage consequences. Internal consistency criteria are satisfied for the scale composed in this way (Cronbachs $\alpha = .66$ - .92).

e) Perceived ease of use. The construct consists of four adapted items from TAM 3 (e.g., "I think the handling of the instructional medium would be clear and understandable for me.") and two further items from the IsoMetrics questionnaire (e.g., "The operating options of the instructional medium support an optimal use of the application."). IsoMetrics was designed for use during the software development process [22]. The focus is set on seven scales, which constitute an operationalization of the seven criteria of the European Committee for Standardization. Here

the scale controllability was used to supplement the items from the TAM. Due to its high standard deviation, one item of the IsoMetrics had to be excluded from the analysis. The remaining four items reached a satisfactory internal consistency of Cronbachs $\alpha = .56$ - .89.

The Post-test on instructional media also contains open questions: "What did you particularly like about the instructional medium you used?", "What would need to be changed in the instruction medium to make the assembly process even easier?", and "Please create a ranking of the instructional media, where 1 is your strongest preference, 2 is your second choice, etc. Please give reasons for your decision."

3) Post-Test – AR application.

The assessment of the usability of the AR application itself was measured by five parameters selected with regard to their fit in terms of early stage evaluation: *(a) perceived usefulness, (b) aesthetic and layout, c) appropriateness of functions,* as well as *d) terminology and terms.* An overview of all scales with example items can be seen in **TABLE II**.

a) Perceived usefulness. To measure perceived usefulness, the same four items were used as in the instructional media post-test. Only the terms were adapted (e.g., "Using the AR application would improve my performance."). Cronbach's alpha showed a satisfactory value of $\alpha = .96$.

b) Aesthetic and layout. In order to comprehensively depict this construct, four items from the "Visual Aesthetics of Websites Inventory – Short (VisAWI-S)" were used in the field of aesthetics [31]. The VisAWI-S records how users subjectively perceive the aesthetics of a graphical interface. The used, short version represents the general aesthetic factor [31]. We adjusted the items in terms of terminology (e.g., "Everything matches within the application") and further added one item from IsoMetrics ("The layout complicates my task processing due to an inconsistent design.") and another from the "Questionnaire for User Interface Satisfaction (QUIS)", which was first published in 1987 to ensure feedback on the font as well [32]. This

TABLE II.   SCALES – AR APPLICATION

Scales	Sources	Number of Items	Final number and Cronbachs α	Example Items
Perceived Usefulness	Technology Acceptance Model (TAM 3) – Perceived Usefulness [30]	4 Items	4 Items Cronbachs α = .96	"Using the AR application would improve my performance."
Aesthetics and Layout	Visual Aesthetics of Websites Inventory (VisAWI-S; [31] Questionnaire for User Interface Satisfaction (QUIS; [32]	4 Items (VisAWI-S) 1 Item IsoMetrics) 1 Items (QUIS)	4 Items (VisAWI-S) 1 Item (IsoMetrics) 1 Items (QUIS) Cronbachs α = .60	"Everything matches within the application." "The layout complicates my task processing due to an inconsistent design."
Appropriate-ness of Functions	IsoMetrics [22]	4 Items	4 Items Cronbachs α = .72	"The information necessary for task processing is always in the right place on the screen."
Terminology and Terms	QUIS [32] ISONORM [22]	4 Items (QUIS) 2 Items (ISONORM)	4 Items (QUIS) 2 Items (ISONORM) Cronbachs α = .65	"On-screen prompts were confusing." "Within the AR application, easily understandable terms, descriptions or symbols (e.g., in masks or menus) are used."
Open Questions	• What did you particularly like about the AR application? • What would have to be changed in the AR application so that the assembly process could be carried out even more easily? • Would you prefer learning via AR classic manuals / manuals? Why?			

composed scale reached an internal consistency of Cronbachs α = .60, which is critical for the analysis of this overall scale.

c) Appropriateness of functions. This scale is based on the Task Adequacy Scale of IsoMetrics and with four items (e.g., "The information necessary for task processing is always in the right place on the screen") reached a Cronbach's alpha of α = .72.

d) Terminology and terms. To illustrate how understandable the terms and instructions used were, we used four items from QUIS (e.g., "On-screen prompts were confusing.") [32]. Furthermore, the transparency of the robot's activities was queried ("The application always informed me about what the robot does."). Two further items (e.g., "Within the AR application, easily understandable terms, descriptions or symbols (e.g., in masks or menus) are used.") for this parameter are taken from the Isonorm questionnaire published in 1993 [22]. Like IsoMetrics, Isonorm is based on the criteria of the European Committee for Standardization and therefore uses the same seven factors. This scale reached in total a Cronbachs alpha of α = .65.

Similar to the instructional media post-test, the post-test for the AR application also contains open questions: "What did you particularly like about the AR application?", "What would need to be changed in AR application to make the assembly process even easier?" Finally, the test persons should decide whether and why they would prefer the AR application to traditional manuals.

D. Analysis

The analysis of the collected data was conducted using SPSS. Open questions and recorded comments were analyzed using MAXQDA software. Since this is still a work in progress, the following is a first insight into the results, with a short outlook on qualitative findings. An inferential statistical comparison of the groups is carried out exploratory by subsuming the individual test conditions to the media

used. Thus, the comparison groups "tablet", "HoloLens" and "touchscreen" are used for the calculations. Due to the small sample, Friedman's ANOVA [25], used as a non-parametric test procedure, provides an insight into existing group differences, which are further investigated with the help of a post-hoc analysis according to Dunn-Bonferroni [25].

V. RESULTS

Section V gives an overview of the first results for pre- and post-test questionnaires as well as results of the open questions on "Instructional Media" and the "AR Application".

A. Pre-Test

The participants have a mean technical affinity of 4.61 (min = 3.4; max = 5.60; SD = 0.71). General control beliefs while dealing with technology is ranging between min = 3.00 to max = 5.75 ($mean$ = 4.73; SD = 0.72) within the sample. Media as PC (n = 7), Laptops (n = 11) and Smartphones (n = 15) are used daily by the majority of the test persons, while HoloLens (n = 12) and the Oculus Rift (n = 13) are used almost never. Only three participants already used the HoloLens before this study.

The evaluation of the Affect Grid (see **Figure 4**) using a one-way ANOVA, shows that there are significant differences between the measurement points (before and after) in the arousal level (F (2, 126) = 6.94, p = .001, partial η^2 = .099) and in the general sensations (F (2, 126) = 30.272, p < .001, partial η^2 = .325) (see **Figure 4**). The diagram shows the comparison of the scales pleasant- unpleasant feelings and sleepiness - arousal before the experiment and after the use of the three different media. The HoloLens shows a tendency towards unpleasant feelings and high arousal, while the use of the touchscreen triggers pleasant feelings and a higher level of sleepiness. However, there are no significant differences between the various conditions (Tablet, HoloLens, and Touchscreen).
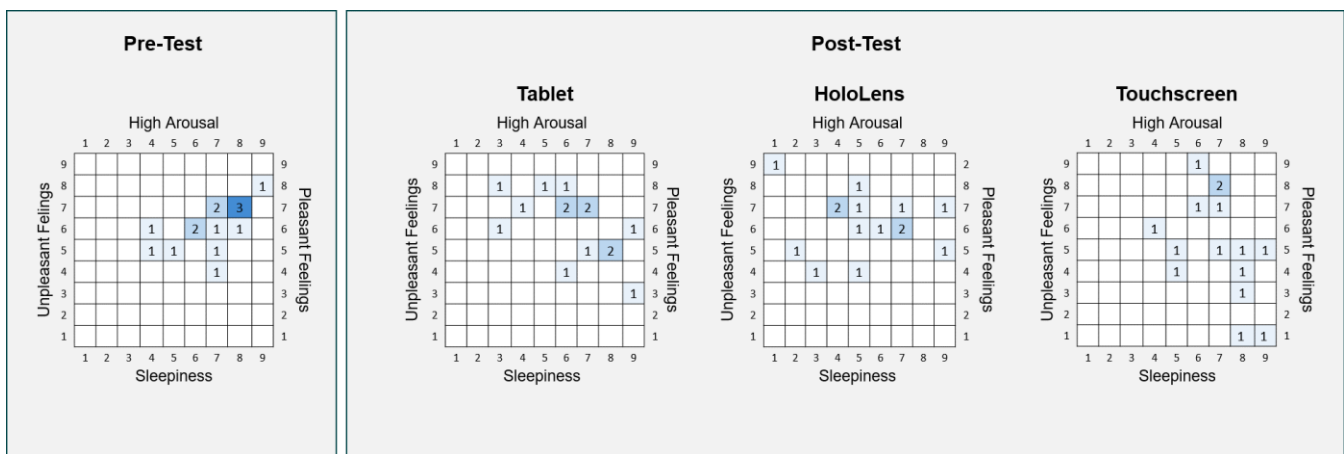

Figure 4.   Affect Grid before and after using Instructional Media

B.   Post-Test – Instructional Media

a) Task-load. Descriptive results of the task load (N = 15; Scale: (0) = "low" to (10) = "high") can be seen in **Figure 5**, where, for each media, the mean of each scale is shown as a percentages. The mean level of frustration over all tasks and media is 41%, and the highest mean level is reached by the HoloLens with 47%. The lowest mean frustration level of 31% is while using the touchscreen.

The participants stated to achieve their goal on a mean of 67% and the highest performance was achieved using the touchscreen (73%). On average, 40% effort was needed to fulfil the assembly task. The mean temporal demand ranges from 36% (touchscreen) to 42% (HoloLens). The highest mean of physical demand was reported using the tablet (53%), the highest mean of mental demand was reported using the HoloLens (61%).
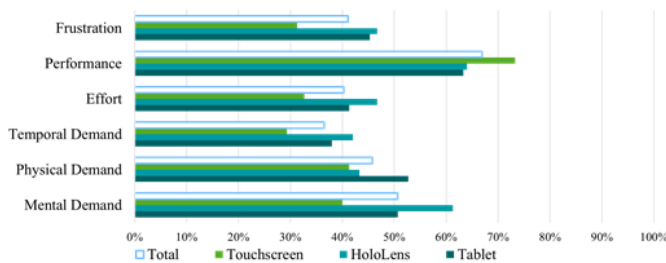

Figure 5.   Task Load of Instructional Media

b) Perceived usefulness. All descriptive statistics of the following scales can be seen in **TABLE III**. Within Friedman's ANOVA, it is always assumed as null hypothesis that there is no difference between the groups. However, the analysis for perceived usefulness shows a statistically significant difference between the groups (x2r (2) = 10.67, p = .005, n = 15). The subsequently performed Dunn-Bonferroni tests with a corrected alpha = .017 show that both the perceived usefulness between tablet and touchscreen differ statistically significantly (z = -2.641, p = .008), as well as the perceived usefulness between HoloLens and touchscreen (z = -2.548, p = .011), indicating that HoloLens and tablet are perceived as less useful than the touchscreen. HoloLens and tablet are not significantly different.

c) Media self-efficacy. Neither mean values nor Friedman's ANOVA show any statistically significant difference between the groups (x^2_r (2) = 4.545, p = .103, n = 15).

d) Perceived Enjoyment. As in the previous scale, neither mean values nor Friedman's ANOVA show a significant difference between the groups with regard to the perceived enjoyment (x^2_r (2) = 2.980, p = .225, n = 14).

e) Perceived ease of use. Both mean values and Friedman's ANOVA indicate a statistically significant difference between the groups with regard to the perceived ease of use of the media (x^2_r (2) = 21.088, p < .000, n = 15). The subsequently performed Dunn-Bonferroni tests with a corrected alpha = .017 show that both the perceived ease of use between tablet and HoloLens differ statistically significant (z = -2.841, p = .005), as well as the perceived

TABLE III. DESCRIPTIVE STATISTICS - INSTRUCTIONAL MEDIA

		Descriptive Statistics				
		N	*Mean*	*SD*	*Min.*	*Max.*
Perceived Usefulness	Tablet	15	3.87	.96	2.50	5.75
	Hololens	15	3.85	1.11	2.00	5.50
	Touchscreen	15	4.87	.93	2.25	6.00
Media Self-Efficacy	Tablet	15	4.80	.95	3.00	6.00
	Hololens	15	4.33	1.51	2.75	6.00
	Touchscreen	15	5.10	.96	3.00	6.00
Perceived Enjoyment	Tablet	14	4.29	.88	2.67	6.00
	Hololens	14	4.38	.66	3.33	5.83
	Touchscreen	14	4.98	.66	3.83	6.00
Perceived Ease-of-Use	Tablet	15	4.60	.60	3.60	5.60
	Hololens	15	3.78	.96	1.80	5.20
	Touchscreen	15	5.17	.59	4.20	6.00

ease of use between HoloLens and touchscreen (z = -3.425, p =.001). Tablet and touchscreen also differ statistically (z = -2.522, p = .012). The results raise an indication that the HoloLens is considered the least easy to use, while the touchscreen reaches the highest value.

Within the ranking of the instructional tools, the touchscreen was selected as a first choice a total of ten times, after that comes the HoloLens with three times, and lastly the tablet with only two times.

Open questions and comments. The evaluation of the verbal expressions and written comments is done by categorizing them into positive and negative comments for each medium. Individual entries are coded several times. In the following, a brief overview of the most frequently mentioned is given (see **TABLE IV**).

Overall, there are 69 positive comments on the media. 33 of these refer to the touchscreen, which is perceived as easy to use, clearly arranged and overall less restrictive compared to other media. With the tablet (n = 18) it is positively evaluated that the animations can be viewed on demand and from different directions. In combination with markers used, some participants find it easier to orientate themselves at the workplace. The HoloLens is 18 times positively evaluated - the intuitive operation and the hands-free working process are mentioned most frequently. In addition, there are 68 negative remarks, 54 of which are verbal and 14 written comments. 38 of these, are related to the HoloLens due to its lack of wearing comfort and limited vision, e.g., because animations overlay the view of actions to be performed.

There are 28 negative comments about the tablet, mainly related to the perceived difficulty of repositioning the tablet arm, resulting in a limited view of the work surface. The touchscreen has only two negative annotations, namely 'the fixation does not provide orientation at the workstation' and 'animations are not displayed on the work surface'.

TABLE IV. WRITTEN COMMENTS AND VERBAL EXPRESSIONS ON INSTRUCTIONAL MEDIA (AR HARDWARE)

		Frequent Categories	Example Statements
Touchscreen	Positive	Ease of Use	"With the touchpad, all necessary information can be grasped most clearly and quickly."
		No Restrictions (e.g., field of view)	"The field of vision is not restricted, I found that somewhat problematic with the glasses." "Especially in comparison to the HoloLens I have the feeling that I am much freer now, because I can operate more easily."
	Negative	Workplace Orientation	"The fixation does not provide orientation at the workstation" "Animations are not displayed on the work surface"
Tablet	Positive	Support on Demand	"It is more relaxed. I look at the monitor when I need help and can then focus on my task."
		Workplace Orientation	"The markers distinguish between different work areas." "The component can be viewed from different angles. Green arrows show where the tablet has to be positioned."
	Negative	Physical Demand	"Tablet must be moved often" "I also need to reach around the tablet's grab arm a little cumbersome here."
		Limited Field of View	"I want to keep looking at the animation, but I also want to work on it at the same time. So I have to use the area on the left side all the time and there is relatively little space to do it."
HoloLens	Positive	Intuitiveness	"The operation of the HoloLens is intuitive and allows an easy interaction with the environment and reacts automatically to the markers, since no manual positioning is required as it follows the eye."
		Hands-free Work Process	"There are no movement restriction, the glasses are intuitive" "Free hands and work area"
	Negative	Wearing comfort	"The HoloLens is rather uncomfortable and interferes with vision." "Glasses slip a lot, but if they're tighter, they hurt. It presses on my nose."
		Overlays	"If you have these glasses on and don't look at the animation, but at what you're doing, you don't see it so well. I'd rather look under my glasses".

C. Post-Test – AR Application

All descriptive statistics of the following scales can be seen in **TABLE V**. The perceived usefulness of the AR application is on a mean of 4.40, which corresponds to an assessment between 'rather agree' and 'agree'. Aesthetics and Layout and Appropriateness of functions (mean = 3.97) corresponds to an assessment of 'rather agree'. Terminology and terms corresponds to an assessment between 'rather agree' and 'agree' with a mean of 4.29.

On the question of whether participants would prefer learning via AR to traditional manuals, 13 out of 15 people say they would prefer AR. Main reasons given are, for example, the active learning process, the small steps, the high degree of interaction, the simplicity of use and the perceived fun. In contrast, comments against included the perceived external control of the technology and the possibility of browsing through manuals at one's own pace.

Open questions and comments. There are a total of 85 positive comments on the AR application. The detailed and vividly visualized animations are mentioned particularly

frequently here (29 entries) and are accompanied by the clearly perceived instructions (13 entries). The fun and excitement in the process (20 entries) and the active, goal-oriented learning process (7 entries) are also mentioned. In the 182 negatively coded expressions, there are often remarks about the lack of correspondence between reality and displayed animations (e.g., in color, degree of detail, or positioning; 34 entries), such as: "it's hard to stay focused while the animation continuously moves in the background". In addition, some animations appeared in places not expected by participants or outside the direct field of vision. Since statements were coded multiple times and often referred directly or indirectly to the specific implementation on the respective medium, a further differentiated quantification was not purposeful. An overview of frequent comments can be seen in **TABLE VI**.

VI. DISCUSSION AND OUTLOOK

The study provides insights into the usability and suitability of different media as assistance systems in collaborative assembly. In a within-subject design, two AR-based media (tablet vs. HoloLens) with the same user interface were tested and compared with a classic medium (touchscreen) to instruct the work steps. The results indicate that although the potential of new assistance technologies is recognized, the classic medium is still judged the easiest to use. Furthermore, the study provides results for the separate evaluation of hard- and software usability, using a tailor-made usability catalogue. According to these results, many problems and weaknesses of the technologies are due to the ergonomics and handling of the hardware, whereas the

TABLE V. DESCRIPTIVE STATISTICS – AR APPLICATION

Descriptive Statistics					
	N	Mean	SD	Min.	Max.
Perceived Usefulness	15	4.40	1.12	2.25	5.75
Aesthetic and Layout	15	3.97	.67	2.83	5.00
Appropriateness of functions	15	4.00	.93	2.40	5.40
Terminology and terms	14	4.29	.71	3.17	5.67

TABLE VI.　　　　WRITTEN COMMENTS AND VERBAL EXPRESSIONS ON AR APPICATION (AR SOFTWARE)

Frequent Categories		Example Statements
Positive	Visualization and Localization	"The animations are close to the point where you should perform the next assembly step."
		"The fact that this is now fully animated makes it even easier to see. So you can really see exactly where the mother has to go now."
	Clear Instructions and Interactivity	"It is more interactive and you learn to do the work step by yourself. You don't have to look at a whole manual beforehand, but are told step by step what to do."
		"You learn more because it's easier to memorize it through taking part. It's also fun."
Negative	Discrepancy between Animation and Reality	"The technology isn't as accurate yet, it doesn't display it perfectly." "I can't see exactly what I'm mounting it on; I'm doing it more intuitively."
	Identification	"If the gears are on top of each other, you can't tell which ones are meant." "You can tell it's the bottom one, but not the one that's on top."
	Localization of Animations	"You don't always know where to look to see the animation." "Because of all the movement, I didn't see that [the animation] was displayed on the far right".

software of the AR application is perceived as very helpful. In general, the methodology and composition of the usability catalogue from inductive and deductive methods was proven to be a successful approach. A high degree of objectivity could be achieved through the questionnaires. The individual results of the scales could be explained in detail by open comments and verbal statements and were thus made comprehensible afterwards. However, the validation of scales in a larger sample should precede further studies.

The results of the evaluation of the instructional media shows that a high level of frustration occurs when processing the task with the HoloLens, which goes hand in hand with a high cognitive demand. Here, possible connections between the ease of use of the media and the perceived cognitive demand could be an interesting starting point for further research. The touchscreen on the contrary causes a low frustration and is evaluated with the highest performance. The tablet's evaluation usually lies between the other media, but shows the highest physical demand. These findings are supported by the assessment of the usability scales, where HoloLens and tablet are rated with a lower usefulness than the touchscreen. In terms of media self-efficacy and perceived enjoyment, there is no difference between the media tested. In future studies, the possible influence of the high technical affinity of the sample on these variables should be clarified. Both the open questions and the ranking support the impression that the touchscreen convinces users with its simple operation. In addition, it becomes clear that the innovative character of the HoloLens is perceived as enjoyable. Above all, the restricted field of vision and the overlapping of animations with reality still poses a problem, which can be prevented, for example, by using the tablet and moving the holder.

In the evaluation of the AR Application, it becomes clear that AR is generally assessed with a relatively high usefulness. This finding is also supported by the open comments. Comparing only the number of positive and negative statements, the impression could arise that the AR application has been increasingly perceived as negative. The

high number of negative comments could be mainly due to the nature of the task. During the thinking aloud method, the test persons are explicitly asked to express all thoughts and ideas. Thereby, it is obvious that the participant first deals with deficits and usability problems - but the reflection and evaluation after the task shows a positive assessment of the application. Many of the comments also contain ideas and suggestions for improvement in order to improve the use of the media.

Especially the clear and small step instructions are perceived as useful. The aesthetics and layout of the application, as well as the appropriateness of functions should be worked on in the further course. It should be considered that images and animations are perceived as helpful, whereas text descriptions are sometimes described as obstructive or misleading.

With regard to the suitability of the assistance technologies presented for the collaborative working scenario, it can be stated that by focusing on the task and the respective medium, the test persons hardly paid attention to the robot and also scarcely made any comments on it. It is problematic that this was not explicitly measured in the questionnaire. The perception of the robot should definitely be investigated more closely in future studies and brought into focus.

The analysis of the Affect Grid shows that, although there is a difference in the level of excitement and pleasure before and after the test, there is no significant difference between the media. However, the descriptive evaluation shows that participants feel increasingly stressed by the HoloLens (axis unpleasant - aroused), whereas the touchscreen shows more frequent entries towards relaxation (axis pleasant - sleepy).

A main restriction of the study refers to the high academic degree and the young average age of the sample, which is accompanied by a comparatively high affinity for technology. In addition, almost all participants are novices in the field of assembly, which severely limits the transferability of study results. Another limitation refers to

the laboratory setting of the study, where no real working conditions (e.g., lighting, noises) occur. The results achieved by such a small number of participants should not be interpreted without caution. Conclusions on the quality of the questionnaire used should not yet be derived, as this requires a larger sample. The items and constructs used here were selected based on the specific use case, which limits the transferability of this selection. Further, we only used already established questionnaires and the thinking aloud method and did not include any other usability instruments (e.g., usability cards).

This paper provides deeper insights into the earlier usability evaluation of the same use case [1]. Special focus was paid to the detailed analysis of the task process in order to align the development of the AR application directly with the use case. This certainly limits the transferability of the results to other use cases. In the meantime, another study has been conducted to test Head Mounted Displays in the field, which has revealed similar results [33]. After the revision of the detected usability problems, following studies should be carried out referring to the effectiveness of the use of AR. Hereby, the influence of AR on the general performance, error rate and satisfaction with the work process should be examined. Furthermore, the influence of AR on the acceptance of robots should be further investigated within the context of collaborative workspaces.

ACKNOWLEDGMENT

REFERENCES

[1] L. Daling, A. Abdelrazeq, M. Haberstroh, and F. Hees, "Usability Evaluation of Augmented Reality as Instructional Tool in Collaborative Assembly Cells," The Twelfth International Conference on Advances in Computer-Human Interactions (ACHI 2019) IARIA, Feb. 2013, pp. 199-205, ISBN: 978-1-61208-686-6.

[2] Acatech - Deutsche Akademie der Technikwissenschaften in Kooperation mit Fraunhofer IML, and equeo, Ed., *"Kompetenzentwicklungsstudie Industrie 4.0: Erste Ergebnisse und Schlussfolgerungen (Industry 4.0 Competence Development Study: First Results and Conclusions),"* München, 2016.

[3] Q. Guo, "Learning in a Mixed Reality System in the Context of Industry 4.0, "*Journal of Technical Education*, vol. 3, no. 2, pp. 92–115, 2015, ISSN 2198-0306.

[4] W. Bauer, M. Bender, M. Braun, P. Rally, and O. Scholtz, "Leichtbauroboter in der manuellen Montage - einfach einfach anfangen: Erste Erfahrungen von Anwenderunternehmen," Frauenhofer IOA, 2016.

[5] S. L. Müller, S. Stiehm, S. Jeschke and A. Richert, "Subjective Stress in Hybrid Collaboration", in vol. 10652, *Social Robotics*, A. Kheddar et al., Eds., Cham: Springer International Publishing, pp. 597–606, 2017, doi: 10.1007/978-3-319-70022-9.

[6] F. Sarodnick and H. Brau, *"Methoden der Usability Evaluation (Methods of Usability Evaluation),"* (2nd. ed). H. Huber, Hogrefe AG, Bern, 2006.

[7] R. Azuma, Y. Baillot, R. Behringer, S. Feiner, S. Julier and B. MacIntyre, "Recent Advances in Augmented Reality", Naval Research Lab, Washington DC, pp. 34-47, 2001, doi: 10.1109/38.963459.

[8] B. Furht, Ed., "Handbook of Augmented Reality," Springer Science & Business Media, 2011, doi: 10.1007/978-1-4614-0064-6.

[9] A. Y. Nee, S. K. Ong, G. Chryssolouris and D. Mourtzis, "Augmented Reality Applications in Design and Manufacturing", *CIRP Annals-manufacturing technology*, vol. 61(2), pp. 657-679, 2012, doi: 10.1016/j.cirp.2012.05.010.

[10] A. Dey, M. Billinghurst, R.W. Lindeman, and J.E. Swan, "A systematic review of 10 years of augmented reality usability studies: 2005 to 2014," Frontiers in Robotics and AI vol. 5:37, 2018, doi: 10.3389/frobt.2018.00037.

[11] A. Little, *"How Automotive Manufacturers are Utilizing Augmented Reality,"* August 2018, retrieved January, 08, 2019 from:https://www.manufacturingtomorrow.com/article/2018/03/how-automotive-manufacturers-are-utilizing-augmented-reality-/11117.

[12] S. Werrlich, D. Austino, A. Ginger, P.-A. Nguyen, and G. Notni, "Comparing HMD-based and paper-based training," Proc. IEEE International Symposium on Mixed and Augmented Reality (ISMAR), IEEE, Munich, Oct. 2018, pp. 134–142, doi: 10.1109/ismar.2018.00046.

[13] M. Akcayır and G. Akcayır, "Advantages and Challenges Associated with Augmented Reality for Education: A systematic review of the literature," *Educational Research Review*, vol. 20, pp. 1-11, 2017, doi:10.1016/ j.edurev.2016.11.002.

[14] X. Wang, S.K. Ong, A.Y.C. Nee, "A comprehensive survey of augmented reality assembly research," Adv. Manuf., vol. 4 (1), pp. 1–22, 2016, doi: 10.1007/s40436-015-0131-4.

[15] S. Makris, P. Karagiannis, S. Koukas, and A. S. Matthaiakis, "Augmented reality system for operator support in human–robot collaborative assembly," CIRP Annals, vol. 65(1), pp. 61-64, 2016, doi: 10.1016/j.cirp.2016.04.038.

[16] G. Michalos, P. Karagiannis, S. Makris, Ö. Tokcalar, and G. Chryssolouris, "Augmented reality (AR) applications for supporting human-robot interactive cooperation," Procedia CIRP, vol. 41, pp. 370-375, 2016, doi: 10.1016/j.procir.2015.12.005.

[17] A. Tang, C. Owen, F. Biocca and W. Mou, "Comparative effectiveness of augmented reality in object assembly," *In Proceedings of the SIGCHI conference on Human factors in computing systems*, pp. 73-80, ACM, April 2003, doi:10.1145/642611.642626.

[18] A. Ullrich and G. Vladova, "Qualifizierungsmanagement in der vernetzten Produktion - Ein Ansatz zur Strukturierung relevanter Parameter (Qualification Management in Networked Production - An Approach to Structuring Relevant Parameters)," *Lehren und Lernen für die moderne Arbeitswelt*, pp. 58–80, GITO, 2015.

[19] J. Nielsen, "Enhancing the explanatory power of usability heuristics," in *Proceedings of the SIGCHI conference on Human Factors in Computing Systems*, pp. 152-158, ACM, 1994, doi: 10.1145/191666.191729.

[20] L. Gabbard, J. E. Swan II, D. Hix, S,-J. Kim and G. Fitch, "Active Text Drawing Styles for Outdoor Augmented Reality: a User-based Study and Design Implications," in *Virtual Reality Conference VR'07, IEEE*, pp. 35-42, 2007, doi: 10.1109/VR.2007.352461.

[21] S. L. Müller-Abdelrazeq, K. Schönefeld, M. Haberstroh, and F. Hees, "Interacting with Collaborative Robots - A Study on Attitudes and Acceptance in Industrial Contexts," in Social Robots: Technological, Societal and Ethical Aspects of Human-Robot Interaction, O. Korn, Ed. Springer International Publishing, pp. 101-117, 2019.

[22] K. Figl, "Deutschsprachige Fragebögen zur Usability-Evaluation im Vergleich (German-language questionnaires for usability evaluation in comparison)," *Zeitschrift für Arbeitswissenschaft*, *4*, pp. 321-337, 2010.

[23] J. Nielsen, "Usability Engineering," London: AP Professional Ltd., 1993.

[24] L. Faulkner, "Beyond the five-user assumption: Benefits of increased sample sizes in usability testing," *Behavior Research Methods, Instruments, and Computers,* 35 (3), pp. 379-383, 2003.

[25] A. Field and G. Hole, "How to design and report experiments," Sage, 2002.

[26] G. Beier, "Kontrollüberzeugungen im Umgang mit Technik: ein Persönlichkeitsmerkmal mit Relevanz für die Gestaltung technischer Systeme (Control Control beliefs in dealing with technology: a personality trait with relevance for the design of technical systems)," *Ph.D. Dissertation*, Humboldt-Universität zu Berlin. Available from GESIS database, Record No. 20040112708.

[27] J. A. Russel, A. Weiss and G. A. Mendelsohn, "Affect grid: a single-item scale of pleasure and arousal", *Journal of personality and social psychology,* 57 (3), pp 493-502, 1989.

[28] M. Thüring and S. Mahlke, "Usability, Aesthetics and Emotions in Human–Technology Interaction", *International Journal of Psychology, 42*(4), pp. 253-264, 2007, doi: 10.1080/00207590701396674.

[29] S. G. Hart, "NASA-Task Load Index (NASA-TLX); 20 Years Later," *Proceedings of the Human Factors and Ergonomics Society 50th Annual Meeting*, Santa Monica: HFES, pp. 904-908, 2006, doi: 10.1177/154193120605000909.

[30] F. D. Davis, "A Technology Acceptance Model for Empirically Testing New End-user Information Systems: Theory and Results," *Ph.D. Dissertation,* Massachusetts Institute of Technology, 1985.

[31] M. Moshagen and M. T. Thielsch, "A Short Version of the Visual Aesthetics of Websites Inventory", *Behaviour & Information Technology*, 32 (12), pp. 1305-1311, 2013, doi: 10.1080/0144929X.2012.694910.

[32] J. P. Chin, V. A. Diehl, and K. L. Norman, "Development of a Tool Measuring User Satisfaction of the Human-Computer Interface", in *Proceedings of the SIGCHI conference on Human factors in computing systems*, pp. 213-218. ACM, 2018.

[33] L. Daling, A. Abdelrazeq, C. Sauerborn, & F. Hees."A Comparative Study of Augmented Reality Assistant Tools in Assembly"; in *International Conference on Applied Human Factors and Ergonomics* (AHFE 2019) Springer, Cham., July 2019, pp. 755-767.

Creating and Evaluating Data-Driven Ontologies

Maaike H.T. de Boer and Jack P.C. Verhoosel

Data Science department, TNO (Netherlands Organisation for Applied Scientific Research),
Anna van Buerenplein 1, 2595 DA, The Hague, The Netherlands
Email: `maaike.deboer@tno.nl` and `jack.verhoosel@tno.nl`

Abstract—**Automatically creating data-driven ontologies is a challenge but it can save time and resources. In this paper, eight data-driven algorithms are compared to create ontologies, four ontologies based on documents and four based on keywords, on three different document sets. We evaluate the performance using three different evaluation metrics based on nodes, weights and relations. Results show that 1) keyword-based methods are in general better than document-based methods; 2) a co-occurrence-based algorithm is the best document-based method; 3) the evaluation metrics give useful insight, but need to be enhanced in future work. It is advised to a) use the created ontologies as a head start in an ontology creation session, but not use the ontologies as created; b) use word2vec to generate an ontology in a generic domain, whereas the co-occurrences algorithm should be used in specific domain.**

Keywords–Ontologies; Machine Learning; NLP; Word2vec; Ontology Learning; F1 score.

I. INTRODUCTION

In the previous decade, data scientists often used either a knowledge-driven or a data-driven approach to create their models / classifiers. In the knowledge-driven approach, the (expert) knowledge is structured in a model, such as an ontology. Some of the advantages of this type of approach are that it is insightful for humans, validated by experts, and it gives a feeling of control. Some of the disadvantages of the knowledge-driven approach are that it takes a lot of dedicated effort to construct the model, it is hard to provide the full model (this is only possible in closed-world domains) and that there might not be one truth. For example, if two experts separately create a knowledge model about the same domain, they probably will come up with different ones, because each expert has his/her own subjective view of important concepts and relations in the domain. Data-driven approaches do not need the dedicated effort from people to construct the model, because an algorithm is used that extracts a model much faster. Disadvantages of data-driven approaches are that the models are often not insightful for humans, they might contain too much noise and might be less 'crisp'. As knowledge-driven and data-driven approaches each have their advantages, a combination of both approaches is worthwhile to use. The field in which ontologies learn from available knowledge using data is named *ontology learning*.

This paper is an extension of our previous paper on this topic [1]. In our previous paper, we proposed an ontology learning methodology that uses existing and new data-driven algorithms to create ontologies based on unstructured textual documents in the agriculture domain. In this paper, we broaden our scope to the pizza domain. Whereas evaluation in the agriculture domain is harder because our document set did not have a matching ground truth ontology, we use the well-known pizza ontology [2] to validate our created initial ontology in that domain. We also use the same pizza document set as presented by Rospoucher et al. [3] to make a comparison possible. Additionally, we extracted another pizza document set based on Wikipedia. In our experiments, eight different ontologies are created for each document set and the performance is evaluated using three evaluation metrics, of which two are those proposed by Rospoucher et al. [3] and our previously proposed relation-based evaluation metric.

In the next section, the related work is described on ontology learning, including open information extraction, and the evaluation of ontologies. In Section III, the methods used to create the different ontologies are explained. Section IV describes the experimental set-up with the datasets, characteristics of the resulting ontologies and our evaluation methodology. Section V contains the results of the evaluation and Section VI contains the discussion of our results. Finally, Section VII contains the main conclusions as well as a description of future work.

II. RELATED WORK

Ontology learning is focused on learning ontologies based on data [4] [5]. One of the most known concepts in ontology learning is the ontology learning layer cake. Starting from the bottom of the cake, the order from bottom to the top of the layer is terms, synonyms, concept formation, concept hierarchy, relations, relation hierarchy, axiom schemata and finally general axioms. Similar to the layered cake, Gillani et al. [6] describe the process of ontology learning by input, term extraction, concept extraction, relation extraction, concept categorization, evaluation, ontology mapping. Ontologies can be learned in three kind of strategies: structured, semi-structured and unstructured data [4]. Examples of these different strategies are: database (structured), HTML or XML (semi-structured) and texts (unstructured). Besides the learning strategies, there are three types of tools available: ontology editing tools, ontology merging tools and ontology extraction tools [7]. In this paper, the focus in on automatically creating ontologies from text, so we focus on unstructured data and ontology extraction tools.

A. OpenIE

Some available ontology extraction tools only focus on the information extraction, up to the relation extraction part of the layered cake. This means that these tools only focus on the creation of triples with a subject, verb or relation, and object. The field that focuses only on the creation of these triples is named Open Information Extraction (OpenIE). According to a recent systematic mapping study by Glauber and Claro [8], the two main steps in OpenIE methods are: 1) shallow analysis or dependency analysis for sentence annotation, such as Part of Speech (PoS) tagging or using the Stanford Dependency Parser; 2) machine learning or handcrafted rules for the extraction of relationship triples. Niklaus et al. [9] make the division between learning-based systems, rule-based systems, clause-based systems and system capturing inter-propositional relationships.

One of the first OpenIE tools is TextRunner [10]. TextRunner tags sentences with PoS tags and noun phrase chunks, in a fast manner with one loop over all documents. TextRunner was followed by WOE (pos and parse), ReVerb, KrakeN, EXEMPLAR, OLLIE, PredPatt, ClausIE, OpenIE4, CSD-IE, NESTIE, MinIE and Graphene among others [8], [9]. All methods use a combination of the two main steps mentioned above. For example, WOE [11] uses machine learning on Wikipedia to learn extraction patterns with PoS tags and dependency parsers. REVERB [12] uses syntactic constraints in the form or PoS-based regular expressions to reduce the number of incoherent and uninformative extractions. OLLIE [13], a follow-up from REVERB, learns from a training set the extraction pattern templates using dependency parsers. It also uses contextual information by adding attribution and clausal modifiers. Most methods often solve the problem of increasing informativeness or decreasing computational complexity [8]. Informativeness links to the number of relevant facts. This is often tackled in the second step, either by increasing the facts using co-reference or transitive inference such as in ClausIE [14], or reducing the facts by using lexical constraints such as in REVERB [12]. Many of the OpenIE tools are not fast and very computational expensive. WOEpos [11] is for example 30 times faster than the original WOEparser, but less accurate.

Recently, deep learning methods, such as the encoder-decoder framework from Cui et al. [15], and the relation extraction method from Lin et al. [16] have been proposed. Although these methods seem fruitful, deep learning seems not yet as overwhelmingly better in all tasks within open information extraction as compared to the field of computer vision [17].

Related to the OpenIE field, query expansion can also be used to find more concepts and relations [18]. This method is often used in the information retrieval field. The most common method is to use WordNet [19]. Boer et al. [20] [21] also use ConceptNet to find related concepts and their relations. Word2vec is also used in information retrieval [22] and ontology enrichment [23] [24].

Concluding, the OpenIE field is quite advanced with many tools and techniques. In our paper, we use some of the state of the art techniques, and use the state of the art from the query expansion field and apply it in the OpenIE field.

B. Ontology Learning tools

One of the oldest methods that use the full ontology learning layered cake is Terminae [25]. Terminae is a method and platform for ontology engineering, and includes linguistic analysis with Natural Language Processing (NLP) tools to extract and select terms and relations, conceptual modeling / normalization (differentiation, alignment and restructuring) and formalization / model checking, with the syntactic and semantic validation.

A second tool is OntoLT [26], which is available as a plugin in Protégé and enables mapping rules. Linguistic annotation of text documents is done using Shallow and CHunk-based Unication Grammar tools (SCHUG) [27], which provide annotation of PoS, morphological inflection and decomposition, phrase and dependency structure. The mapping rules can then be used to map the ontologies or the document into one ontology.

A third tool is Text2Onto [28]. Text2Onto uses GATE to extract entities. GATE [29] has a submodule named AN-NIE that contains a tokeniser, sentence splitter, PoS tagger, gazetteer, nite state transducer, orthomatcher and coreference resolver. Several metrics, such as Relative Term Frequency (RTF), Term Frequency Inverted Document Frequency (TF-IDF), Entropy and the C-value/NC-value are used to assess the relevance of a concept. The relations between concepts are found with WordNet, Hearst patterns, and created patterns in JAPE. With the Probabilistic Ontology Model, the tool should be robust to different languages and changing information. According to Zouaq et al. [30], Text2Onto generates very shallow and light weight ontologies.

A fourth tool is Concept-Relation-Concept Tuple based Ontology Learning (CRCTOL) [31]. CRCTOL uses the Stanford PoS tagger and the Berkeley parser to assign syntactic tags to the words. They use a Domain Relevance Measure (DRM), a combination of TF-IDF and likelihood ratio, to determine the relevance of a word or multi-word expression. LESK and VLESK are used for word sense disambiguation. Hearst patterns, relations in WordNet and created patterns with regular expressions are used to find relations with the relevant terms. According to Gillani et al. [6], CRCTOL only creates general concepts and ignores whole-part relations, the ontology is not the comprehensive and accurate representation of a given domain and it is time-consuming to run the tool, because it does full-text parsing.

A fifth tool is CFinder [32], which is created to automatically find key concepts in text. They use the Stanford PoS tagger, a dictionary lookup for synonym finding, stopword removal, and combination of words to also have dependent phrases as concepts. The key concepts are then extracted using a rank-based algorithm that uses the TF (Term Frequency) and a domain specific DF (Document Frequency) as weight. The paper stops at the key concept extraction and does not go further with determining relations.

A sixth tool is OntoUPS [33]. OntoUPS uses the Stanford dependency parser, and learns an Is-A hierarchy over clusters of logical expressions, and populates it by translating sentences to logical form. It uses Markov Logical Networks (MLNs) for that.

A seventh tool is OntoCMaps [30], which uses the Stanford PoS tagger and dependency parser to extract concepts. It uses several generic patterns to extract relations.

A eighth tool is Promine [6]. Promine uses tokenization, stop word filtering, lemmatization, and term frequency to create a set of key words. Wordnet, Wiktionary and a domain glossary (AGROVOC) are used for concept enrichment. The relevance, or term goodness, is calculated with the information gain, which combines the entropy and conditional probability. The concepts are filtered using the information gain, path length and depth of concepts.

Besides, FRED [34] transforms text to LinkedData, using theory from combinatory Categorial Grammar, Discourse Representation Theory, Frame Semantics and Ontology Design Patterns.

Additionally, Tiddi et al. [35] use LinkedData as a basis to create an ontology. They use a dependency analysis to extract entities and the TF-IDF frequency to filter patterns. An entity discovery is done using web queries.

Bendaoud et al. [36] have a semi-automated process in which an ontology is constructed from document abstracts. The formal concept analysis framework is extended to a relational

concept analysis to find links and infer relations between concepts.

Related to ontology learning, Mittal et al. [37] recently combined knowledge graphs and vector spaces into a VKG structure. In that way, both a smart inference from the knowledge graphs and a fast look-up from the vector spaces are combined. This method, however, does not automatically create a new ontology from text documents.

Also, deep learning is used in knowledge graphs. Schlichtkrull et al. [38] propose a Graph Convolutional Network to predict missing facts and missing entity attributes. This method can, thus, also not create an ontology from a set of documents, but is able to enrich an existing ontology.

Concluding, many ontology learning tools are already available and many use OpenIE techniques first and build upon those. We use some of the OpenIE used in the tools as state of the art for our algorithms.

C. Evaluating ontologies

Brank et al. [39] state that most approaches to evaluate ontologies can be placed in one of the following categories:

- Golden Standard: compare to "golden standard"
- Application-based: use in application and evaluate results
- Data-driven: involve comparisons with a data source
- Assesment by humans: human evaluation based on a set of predefined criteria, standards, and / or requirements

Hlomani et al. [40] also use these approaches in their survey, and state the advantages and disadvantages of each approach. We focus on the disadvantages of the approaches first. In the golden standard, the main disadvantage is the evaluation of the golden standard and the performance is highly dependent on the quality of the golden standard. In the application-based approach, the disadvantage is generalizability: what might be a good ontology in one application does not have to be a good one in another. The application-based approach is also only applicable for a small set of ontologies. The main disadvantage of the data-driven approach is that the domain knowledge is assumed to be constant, which is not the case. Finally, the disadvantage of the human assessment is subjectivity.

In this paper, we focus on the data-driven evaluation a well as a golden standard for one of our domains. In the data-driven approach the ontology is often compared against existing data about the domain. Many papers on this topic focus on some kind of coverage of the domain knowledge within the ontology [41]–[44]. For example, Brewster et al. [44] compare extracted terms and relations from text with the concepts and relations in an ontology. They use a probabilistic model to determine the best ontology for a certain domain. OOPS! focusses on pitfalls in ontologies and target newcomers and domain experts [45].

Besides the categories, ontologies can be evaluated on different levels. These levels are defined differently in different papers. Brank et al. [39] divide the levels in lexical, hierarchical, other semantic relations, context, syntactic, and structure. They link the categories and the levels in a matrix, in which the human assessment is the only category which evaluates on all levels. The data-driven approach can only evaluate on the first three levels. The distinction of Burton et al. [46] is syntactic, semantic, pragmatic and social. Gangemi et al. [47]

use the distinction between structural, functional and usability-profiling. Burton et al. [46] use lawfulness, richness, interpretability, consistency, clarity, comprehensiveness, accuracy, relevance, authority, and history. Lozano et al. [48] even use a three-level framework of 117 criteria. Hlomani et al. [40] make the distinction between ontology quality and ontology correctness views on ontology evaluation. For ontology quality, they focus on computational efficiency, adaptability and clarity. Ontology correctness uses accuracy, completeness, conciseness and consistency. Tiddi et al. [35] use the F-measure and precision and recall to evaluate ontology correctness by checking 1) whether attribute values are correctly extracted and 2) how much of the existing knowledge is extracted (based on DBpedia). Rospocher et al. [3] use the same performance metrics to compare an ontology with a list of automatically extracted keywords. Recently, Mcdaniel et al. [49] introduced the DOORS framework in which ontologies can be ranked by using syntactic, semantic, pragmatic and social quality metrics.

III. METHOD

In this paper, we create a taxonomy or concept hierarchy, and we do not include the top two layers of the layered cake (domain, range and axioms / generic rules). This means that this is a first step towards an ontology, but although we have created an owl file, it is not as rich as a real ontology with domains and rules. Figure 1 shows an overview of the methods used to create the ontologies. From each article first the plain text is extracted from the PDF. On these plain texts sentence splitting is used, as well as tokenizing, removing non-ascii and non-textual items and non-English sentences as pre-processing. With these pre-processed texts the ontologies named Hearst, Co-oc and OpenIE (explained below), an our previously proposed Dep++ method [1] are created. The ontologies are named after the algorithm they are made with.

To create the keyword-driven ontologies, keywords have to be extracted. Several keyword extraction methods exist. Instead of the keyword extraction method by Rospoucher et al. [3], which uses KX [50] to get an ordered list of keywords, we combine the Term Frequency (TF) and the term extraction method from Verberne et al. [51]. The standard Wikipedia corpus from the paper is used as background set. We combined the keywords of the two sets and manually deleted all subjectively determined non-relevant terms, resulting in the following set of 12 keywords for Agri: *Data, Food, Information, Drones, Agriculture, Crop, Technology, Agricultural, Production, Development, Farmers, Supply Chain*. And for the pizza case we use the following 13 keywords: *cheese, pizza, sauce, peppers, chicken, mozzarella, onion, tomato, pepperoni, mushroom, bacon, olive, italian* These keywords are used to create the Word2vec, WordNet and ConceptNet ontologies as well as the combined ontology of these three.

A. Hearst

Hearst patterns [52] can be used to extract hyponym relations, represented in an ontology as a 'IsA' relation. An example is 'Vegetable' is a hyponym of 'Food'. In unstructured texts, hyponyms can be spotted using the lexical structures 'NP, such as NP', or 'NP, or other NP', where NP is a noun phrase. These patterns are used to create an ontology with 'IsA' relations.
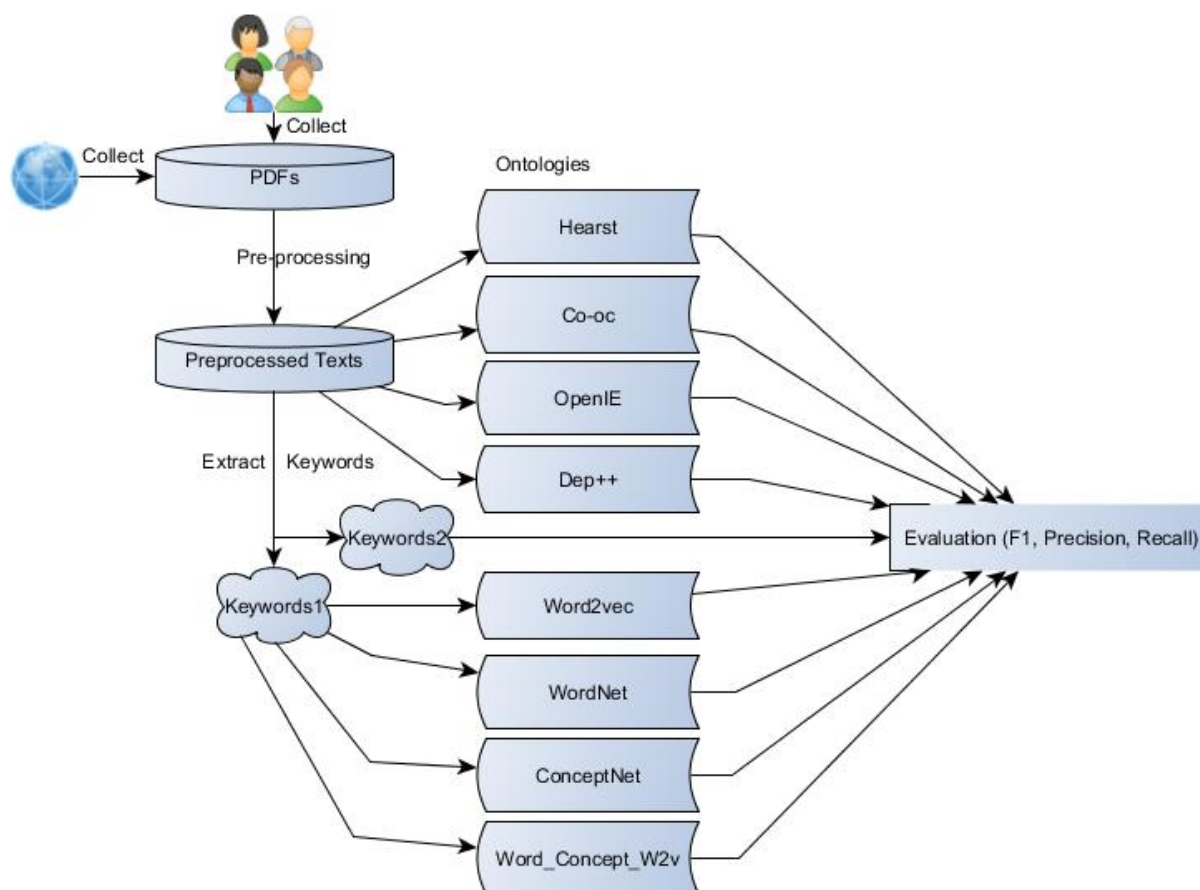
Figure 1. Overview of the methods to create the ontologies

B. Co-oc

Co-occurrences can extract all type of relations, because the number of times words co-occur with each other, for example in the same sentence, are counted [53]. We calculated the set of pairs of different words that co-occur in the sentences of the document set with a maximum distance of 4 words. Therefore, the N-gram generator of the CountVectorizer module of the Scikit-learn package [54] is used and the set is cleaned with the built-in English stopword list. As this set of co-occurring pairs of words will be very large, we further pruned the set using a threshold on the minimum number of times a pair of words co-occurs. This threshold is defined as a percentage of the maximum number of times a co-occurring pair of words is found. In the experiments, this number is set to 10. This number is based on experimentation with several values (ranging from 1 to 50) and overall performance seems best with 10 in our case. The ontology based on these co-occurring pairs of words will have only one vague 'co-occurrence' relation, indicating that the words that co-occur with each other in the document set. The specific type of relation is not determined.

C. OpenIE

The Open Information Extraction tool (OpenIE) is created by the CoreNLP group of Stanford [55]. The tools from the Stanford CoreNLP group are one of the most used tools in the NLP field. The OpenIE tool provides the whole processing chain from plain text through syntactic analysis (sentence splitter, part-of-speech tagger, dependency parser) to triples (object - relation - subject). The extracted relations are often the verbs in the sentence, and this results in triples, multiple word concepts, and many different relations.

D. Dep++

Similar to OntoCMaps [30], syntactic patterns are used to enhance the the Stanford Dependency Parser [56]. The algorithm consists of the following steps: a. Take each document in the corpus and generate sentences based on NLTK tokenization; b. Consider only sentences with more than 5 words which pass through the English check of the Python langdetect package; c. Parse each sentence through the Stanford DepParse annotator to generate Enhanced++Dependencies; d. Replace every word in the Enh++Dep by its lemma as produced by the Stanford POS tagger to consider only singular words; e. Then, generate a graph with a triple <governor,dependency,dependent> for each enhanced++dependency and apply the following transformation rules to the it:

- 1: Transform compound dependencies into 2-word concepts using rule: if $(X, compound, Y)$ then replace X with YX and remove Y
- 2: Enhance subject-object relations based on conjunction dependencies using rule: if $(X, nsubj, Y)$

and $(X, dobj, Z)$ and $(X, conj\_and, X')$ then add $(X', nsubj, Y)$ and $(X', dobj, Z)$

Finally, apply language patterns to derive triples from the dependency graph:

- pattern 1: if $(X, amod, Y)$ then add triple $(YX, subClassOf, X)$
- pattern 2: if $(X, compound, Y)$ and $(X \text{ is } NN \text{ or } NNS)$ then add triple $(YX, subClassOf, X)$
- pattern 3: if $(X, nsubj, Y)$ and $(X, dobj, Z)$ then add triple (Y, X, Z)

This algorithm yields an ontology that is similar to the OpenIE ontology, but should have less triples and thus less noise in it in terms of NLP-based constructs.

E. Word2vec

The first keyword-based method explained here is Word2vec. Word2vec is a group of models, which produce semantic embeddings. These models create neural word embeddings using a shallow neural network that is trained on a large dataset, such as Wikipedia, Google News or Twitter. Each word vector is trained to maximize the log probability of neighboring words, resulting in a good performance in associations, such as *king - man + woman = queen*. We use the skip-gram model with negative sampling (SGNS) [57] to create a semantic embedding of our agriculture documents. With the keywords, we search for the top ten most similar words and add a 'RelatedTo' relation between the keyword and this most similar word. This process is repeated for all most similar words.

F. WordNet

WordNet is a hierarchical dictionary containing lexical relations between words, such as synonyms, hyponyms, hypernyms and antonyms [58]. It also provides all possible meanings of the word, which are called *synsets*, together with a short definition and usage examples. WordNet contains over 155,000 words and over 206,900 word-sense pairs. We use the keywords to search in WordNet. We select the first synset (the most common), extract the 'Synonym' and 'Antonym' relations, and use these to create our ontology.

G. ConceptNet

ConceptNet (version 5) is a knowledge representation project in which a semantic graph with general human knowledge is build [59]. This general human knowledge is collected using other knowledge bases, such as Wikipedia and WordNet, and experts and volunteers. Some of the relations in ConceptNet are *RelatedTo*, *IsA*, *partOf*, *HasA*, *UsedFor*, *CapableOf*, *AtLocation*, *Causes*, *HasSubEvent*, *CreatedBy*, *Synonym* and *DefinedAs*. The strength of the relation is determined by the amount and reliability of the sources asserting the fact. Currently, ConceptNet contains concepts from 77 language and more than 28 million links between concepts. We use the keywords to search (through the API) in ConceptNet and extract all direct relations to create the ontology.

H. Word_Concept_W2v

This method takes the union (all relations) from the keyword-based methods WordNet, ConceptNet and Word2vec. This is, thus, a self created algorithm that combines the other three algorithms.

IV. EXPERIMENTAL SET-UP

A. Document Sets

In our experiment, three different document sets are used, of which two are dedicated to pizzas and one is focused on our application domain of Agriculture. The document sets are described below.

a) PizzaMenus: The first pizza document set is the one created by Rospocher et al. [3]. This document set consists of 50 online available pizza restaurant menus, together approximately 22,000 words. The pizza types, ingredients, types of crust and details on sizes and prices are described, but also other information about beverages and other types of food such as sandwiches.

b) PizzaWiki: The other pizza document set is based on the information on Wikipedia. The original description of pizza is used, as well as all descriptions of pizza varieties and cooking varieties that were present as a box in the pizza description (as of date July 4th, 2019). This resulted in a set of 45 documents about pizza.

c) Agriculture: Our experts collected 135 articles on the Agriculture domain, including Agrifood, Agro-ecology, crop production and the food supply chain.

B. Ontologies

The methods used to create the ontologies are explained in the previous section. The keywords for the pizza domain are not dependent on the pizza document set, because they are partly manually created, so the ontologies for both the PizzaMenus and PizzaWiki is the same. Table I shows the number of classes in the ontologies and some examples of the relations with the word 'pizza' in both the PizzaMenus and PizzaWiki docset. Table II shows similar information for the Agriculture document set, with the word 'Agriculture'. These are randomly picked words from the extracted triples, just to give an idea of some of the words extracted by each of the methods.

TABLE I. INSIGHTS IN THE PIZZA ONTOLOGIES.

OntologyName	#Classes	#Relations	RelationPizza
Hearst$_{menus}$	4	2	tomato sauce specialty
Co-oc$_{menus}$	99	369	crust, bbq, onions, fresh
OpenIE$_{menus}$	840	887	dominos, sesame seeds, beef, anything
Dep++$_{menus}$	1218	916	dough, topping, sauce, you
Hearst$_{wiki}$	153	110	pizza chains, hawaiian pineapple, prezzo
Co-oc$_{wiki}$	113	164	crust, italian, topping, city
OpenIE$_{wiki}$	5690	8160	baked, popular, see food, topped
Dep++$_{wiki}$	3458	2953	frozen, crust, deep-fried, cheese
Word2vec	46	274	garlic, sauce, pepper, tomato
WordNet	54	53	pizza pie
ConceptNet	109	111	pepperoni, hamburger, deliver, oven
Word_Concept_W2v	171	438	oven, mushrooms, olives, green peppers

TABLE II. INSIGHTS IN THE SMARTGREEN ONTOLOGIES.

OntologyName	#Classes	#Relations	RelationAgriculture
Hearst	7523	7906	sector, yield forecasting, irrigation
Co-oc	1049	132,068	food, woman, adopt, production
OpenIE	280,063	535,380	sustainability, they, vision, water use
Dep++	178,338	205,251	sustainable, industrial, we, climate-smart
Word2vec	234	264	farming, biofuel, horticulture, innovation
WordNet	113	116	agribusiness, factory farm, farming
ConceptNet	203	213	farm, farmer, class, agribusiness
Word_Concept_W2v	491	593	agribusiness, farming, farm, horticulture

C. Evaluation

In order to evaluate the performance of the algorithms, it would be best to have a golden standard ontology for the domain that can be used as ground truth. Then, the challenge is to determine how close the created ontologies are to this golden standard ontology in terms of the number of concepts and relations in the created ontology that are also in the golden standard ontology.

Since we do not have a golden standard ontology in our agriculture case, a set of keywords is generated from the input document set using the KLdiv method and taken as ground truth. KLdiv is a proven good method for keyword extraction and therefore we assume that it generates keywords that are close to the ground truth with respect to the concepts that need to be present in the ontology. Then, the assumption is that the semantic quality of a created ontology is better if a keyword is present as concept in the ontology. It might be one of the best data-driven methods, but obviously not as good as human ground truth. The advantage is, though, that the number of keywords can be set. These evaluation-keywords are slightly different from our partly manually selected keyword set. For the pizza document sets, we calculate both the performance based on the keywords and the performance based on the pizza ontology [2] that is considered to be a golden standard ontology.

Although we cannot guarantee this is the best method to test the full range of the capabilities and performance of the algorithms, the three different datasets and the different number of keywords give a sense of the diversity in the results and the performance of the algorithms.

To evaluate the created ontologies, three different metrics to calculate a F1 score are used, which is based on a precision and a recall score. The first two metrics are based on the formulas proposed by Rospocher et al. [3] and the last metric also takes the relations between concepts into account.

D. Node-based F1

$$Prec_{node} = \frac{k \in K_{correct}}{\#k \in Onto}$$

$$Rec_{node} = \frac{k \in K_{correct}}{\#k \in K} \qquad (1)$$

$$F1_{node} = 2 * \frac{(Rec * Prec)}{Rec + Prec}$$

where k is a keyword, which can be found in the set of correct keywords ($K_{correct}$), the total set of extracted keywords (K) and in the ontology ($Onto$) to be evaluated.

E. Weighted Node-based F1

$$Prec_{wnode} = \frac{k \in K_{correct}}{\#k \in Onto}$$

$$Rec_{wnode} = \frac{\sum(rel_{kcorrect}) \in K_{correct}}{\sum(rel_k) \in K} \qquad (2)$$

$$rel_k = F1_{wnode} = 2 * \frac{(Rec * Prec)}{Rec + Prec}$$

where k is a keyword, which can be found in the set of correct keywords ($K_{correct}$), the total set of extracted keywords (K) and in the ontology ($Onto$) to be evaluated. $rel_{kcorrect}$ is the sum of the relevance scores in $K_{correct}$ and rel_k is the

sum of the relevance scores in K. The relevance scores are determined through the KLdiv weights.

F. Relation-based F1

$$Prec_{rel} = \frac{\#r \in R \text{ with } k \in K}{\#r \in R}$$

$$Rec_{rel} = \frac{\#k \in K \text{ found in } R}{\#k \in K} \qquad (3)$$

$$F1_{rel} = 2 * \frac{(Rec * Prec)}{Rec + Prec}$$

where k is keyword in set of Keywords (K), r is relation in set of Relations (R). The set of selected items is thus the set of relations R (precision), and the set of relevant items is thus the set of keywords K (recall).

V. RESULTS

A. PizzaMenus and PizzaWiki

Figure 2 shows for each ontology the overall quality based on the F1 score of the keywords for 15, 30, 50, 100, 150 and 200 keywords, Figure 3 for the weighted version and Figure 4 for the relation-based version.

The results show that in the node based F1 methods word2vec seems to be the best method until 100 keywords. Co-oc is the second best. The pizza menus document set gives slightly better results compared to the PizzaWiki document set. In the relation-based evaluation, the combined keyword-based methods word_concept_w2v scores high, as well as the Co-oc Wiki. The Wiki document set seems slightly better compared to the Menus document set with this evaluation metric.

Table III shows the results of the node-based F1 scores based on the pizza ontology [2]. The concepts of the pizza ontology are used as keywords. The pizza ontology can contain multiple words in one concept, but we define a correct concept in the created ontology to be evaluated has to match at least one word in a golden keyword. The keyword weight is set to the number of edges of the keyword in the golden ontology.

The results of the table show that Co-oc of the Menus document set is the best method for the normal F1, whereas word2vec is the best with the weighted F1 metric. This is similar to the results with the keywords.

TABLE III. F1 SCORES OF THE PIZZA ONTOLOGIES BASED ON THE GROUND TRUTH

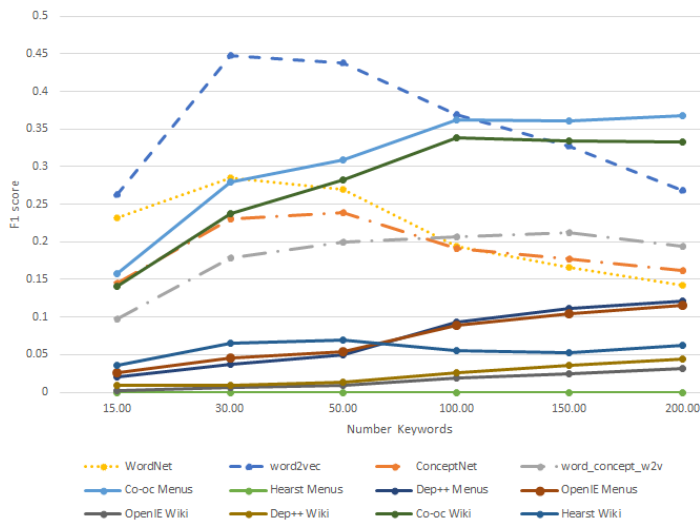| OntologyName	F1 node	F1 weighted node
Hearst$_{menus}$	0.019	0.022
Co-oc$_{menus}$	0.311	0.369
OpenIE$_{menus}$	0.043	0.045
Dep++$_{menus}$	0.051	0.052
Hearst$_{wiki}$	0.067	0.081
Co-oc$_{wiki}$	0.211	0.244
OpenIE$_{wiki}$	0.013	0.013
Dep++$_{wiki}$	0.022	0.022
Word2vec	0.283	0.403
WordNet	0.164	0.260
ConceptNet	0.190	0.239
Word_Concept_W2v	0.177	0.203

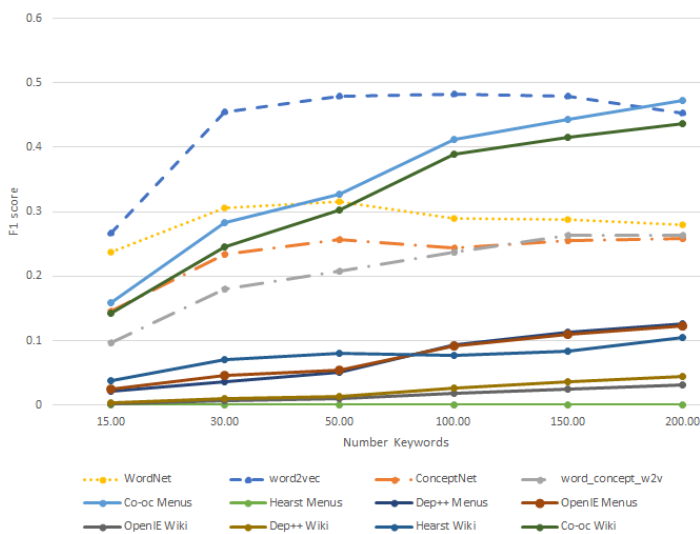Figure 2. Node-based F1 score for Pizza datasets

Figure 3. Weighted Node-based F1 score for Pizza datasets
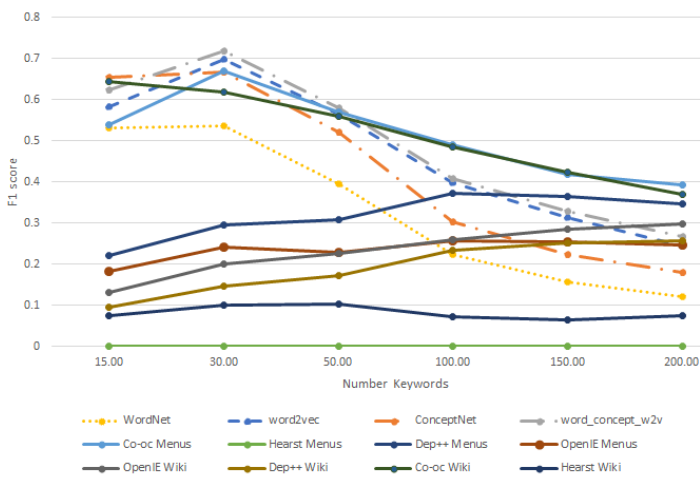

Figure 4. Relation-based F1 score for Pizza datasets

B. Agriculture

Figure 5 shows for each ontology the overall quality based on the F1 score of the keywords for 15, 30, 50, 100, 150 and 200 keywords, Figure 6 for the weighted version and Figure 7 for the relation-based version.

The results show that in the node-based F1 scores, the single keyword-based methods, especially WordNet, are better than the document-based methods. After 50 keywords, the Co-oc method is the highest scoring method and performance becomes twice as good compared to the other methods. The difference between the normal node-based method and the weighted method is mainly visible in the WordNet score, which declines less in the weighted version. The trend of the lines in the relation-based evaluation is different from the node-based evaluation methods. Co-oc is scores highest for all number of keywords. The keyword-based methods perform better compared to the node-based evaluation and with more keywords even outperform some of the keyword-based methods. The combined word_concept_w2v method is in the relation based evaluation better compared to the single methods.
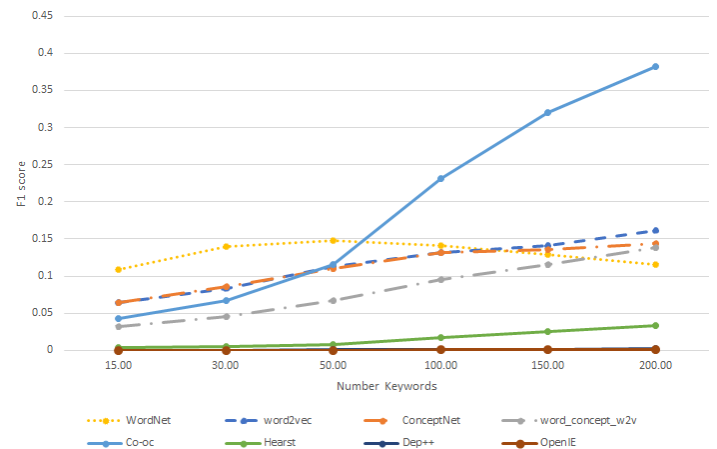

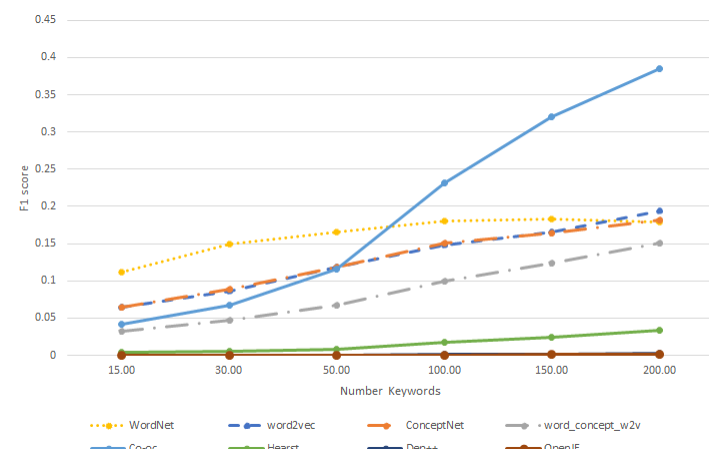Figure 5. Node-based F1 score for SmartGreen dataset

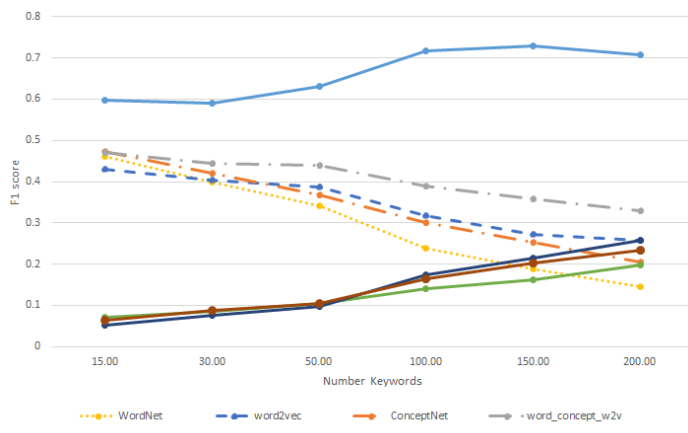Figure 6. Weighted Node-based F1 score for SmartGreen dataset

Figure 7. Relation-based F1 score for SmartGreen dataset

VI. DISCUSSION

In this section, we discuss the results of our experiments by comparing the document sets, algorithms and evaluation methods with each other.

First, when we compare the results between the Pizza-Menus and PizzaWiki docsets, we see that only Hearst is better for PizzaWiki, because only 2 matches were found in Menus. For all other algorithms PizzaMenus is better compared to PizzaWiki. For example in Co-oc, 28 matches are found in PizzaMenus, whereas 17 are found in PizzaWiki. We expected this to be the other way around, because usually Wikipedia descriptions contain better natural language phrases than menus. We conclude therefore, that the menus do not contain a lot of correct English sentences, but do contain the correct keywords. Overall, Word2Vec is best when 100 evaluation-keywords or less are used. Above 100, Co-oc becomes best and remains at the same level with increased number of evaluation-keywords. Another interesting result is that WordNet is not better then Word2Vec. A reason can be that WordNet adds a few synonyms to the ontology that are not keywords and thus decreases precision and F1.

Second, when we compare the different evaluation metrics, we see that for both the Pizza and Agriculture docsets, the curves for node-based and weighted node-based F1 scores are very similar. In absolute terms, the weighted node-based F1 score is slightly better. Comparing our results of the PizzaMenus docset to the results of Rospocher et al. [3], they report an F1 score of 0.17 and a weighted F1 of 0.25 on the PizzaMenus. They enrich the keywords that are extracted with WordNet concepts, which is similar to our WordNet algorithm. Fortunately, we can see that our F1 score of 0.18 and weighted F1 score of 0.26 is almost equal. On the other hand, the curves of our relation-based F1 score look different. When the number of evaluation-keywords increases the F1-score decreases for most of the algorithms, except for Co-oc in the Agriculture docset. This is due to a stronger decrease of the recall with increasing number of evaluation-keywords. An explanation for this effect might be that the counters in the relation-based precision and recall definitions differ in contrast to the node-based precision and recall definitions.

Third, when we compare the F1 scores of the Pizza document sets with the Agriculture docset, we see that in the

Pizza document sets both Word2Vec and Co-oc perform best, while in the Agriculture docset Co-oc outperforms the other algorithms. Specifically, Co-oc becomes best with increased number of evaluation-keywords above 60, which is lower than the 100 evaluation-keywords with the Pizza docsets. So, the main difference between the results of the Pizza docsets and the Agriculture docset is the curve of the Word2Vec algorithm. This can be explained as follows. In principle, the F1 score of every algorithm follows a parabolic form with a peak. The peaks of the curves of the keyword-based algorithms appear in the range of 15 to 200 evaluation-keywords for the Pizza docsets but not for the Agriculture docset. This is dependent on the number of concepts in the generated ontologies and thus on the size of the docset. In general, we can conclude that the peak in the F1-score appears when the number of evaluation-keywords is close to the number of concepts in the ontology that is evaluated.

Fourth, when looking at the purely NLP-based algorithms OpenIE and Dep++, we see that both do not perform very well. The number of concepts and relations generated by Dep++ are considerably less compared to those generated by OpenIE. The number of matching concepts is approximately the same for both algorithms. Therefore, we conclude that the Dep++ algorithm generates a smaller and slightly better ontology, but that this ontology still contains a lot of 'noise' in terms of non-relevant concepts and relations.

Finally, we compare keyword-based algorithms with the document based algorithms. In general the keyword-based algorithms are better than the document-based ones. The only exception is the Co-oc algorithm that outperforms all other algorithms. An explanation of this effect can be that apparently the main keywords of the docset often appear close to each other and are, therefore, part of the Co-oc generated ontology. Despite that fact, the keyword-based methods have the advantage that 1) they are based on general knowledge bases and thus no domain-specific documents have to be collected and 2) in an expert session to build up a domain ontology only a set of keywords have to be generated and agreed upon. A disadvantage is that when a topic becomes very specific, common knowledge bases have very sparse information, whereas domain-specific documents might provide more information. This is also visible in the number of classes and relations found for the different methods. We advise to start with building a domain ontology based on generic topics that are present in the domain using the keyword-based algorithms, preferably Word2Vec. When domain-specific topics need to be added, it is better to use the Co-oc algorithm based on a docset of specific domain documents.

We conclude this discussion with the following issues:

- For the pizza domain, the use of a set of menu documents is feasible, but for most domains this is not possible and it is, therefore, better to use Wikipedia as a basis for finding documents or articles about the domain.
- The current definitions of precision and recall are mostly based on concepts and partly on relations. Thus, a better definition that fully takes relations account is needed. This is one of our future work items.
- With the current results and precision and recall definitions we can achieve F1 scores of up to around 0.7.

Using more strict definitions of precision and recall, this will most probably drop down below 0.5. The question is whether this performance results in domain ontologies that are acceptable for domain experts to function as a head start for an expert meeting. Future experiments in which domain experts are involved in the evaluation phase are therefore necessary.

VII. Conclusion

Creating ontologies is takes time and effort. In this paper, we examined whether we can create data-driven ontologies based on a set of documents to start an ontology-creation session with a head start. In our experiments, we compare 8 different data-driven algorithms, four based on the documents themselves and four based on extracted keywords. We use two pizza document sets and one agriculture document set to generate ontologies with these algorithms. Finally, we use three different evaluation metrics to compare performance in terms of precision, recall and F1-score.

The results show that the keyword-based methods in general outperform the document-based methods. The only exception is Co-oc. Based on these results, we suggest to use the Word2Vec method in a domain with general topics and shift to Co-oc for specific topics for which no information is present in common knowledge bases. The different evaluation metrics show similar trends. The advantage of the relation-based metric is that the relation is taken into account. Future work should be done to further optimize and validate this metric in order to define correct relations of the ontologies. Another topic of future work is to qualitatively validate the ontologies, for instance using the layered ontology metrics suite for ontology assessment with syntactic, semantic, pragmatic and social quality criteria. Using human evaluators it can be verified whether the current performance is high enough to be valuable to use as a head start in an ontology creation session.

Acknowledgment

References

[1] M. de Boer and J. Verhoosel, "Creating data-driven ontologies: An agriculture use case," in ALLDATA 2019: the Fifth International Conference on Big Data, Small Data, Linked Data and Open Data, Valencia, Spain 24-28 March 2019, 52-57, 2019.

[2] "Pizza.owl," https://protege.stanford.edu/ontologies/pizza/pizza.owl, accessed: 2019-07-25.

[3] M. Rospocher, S. Tonelli, L. Serafini, and E. Pianta, "Corpus-based terminological evaluation of ontologies," Applied Ontology, vol. 7, no. 4, 2012, pp. 429–448.

[4] P. Cimiano, A. Mädche, S. Staab, and J. Völker, "Ontology learning," in Handbook on ontologies. Springer, 2009, pp. 245–267.

[5] C. A. Brewster, "Mind the gap: Bridging from text to ontological knowledge," Ph.D. dissertation, University of Sheffield, 2008.

[6] S. Gillani and A. Kő, "Promine: a text mining solution for concept extraction and filtering," in Corporate Knowledge Discovery and Organizational Learning. Springer, 2016, pp. 59–82.

[7] J. Park, W. Cho, and S. Rho, "Evaluating ontology extraction tools using a comprehensive evaluation framework," Data & Knowledge Engineering, vol. 69, no. 10, 2010, pp. 1043–1061.

[8] R. Glauber and D. B. Claro, "A systematic mapping study on open information extraction," Expert Systems with Applications, vol. 112, 2018, pp. 372–387.

[9] C. Niklaus, M. Cetto, A. Freitas, and S. Handschuh, "A survey on open information extraction," arXiv preprint arXiv:1806.05599, 2018.

[10] A. Yates and et al., "Textrunner: open information extraction on the web," in Proceedings of Human Language Technologies: The Annual Conference of the North American Chapter of the Association for Computational Linguistics: Demonstrations. Association for Computational Linguistics, 2007, pp. 25–26.

[11] F. Wu and D. S. Weld, "Open information extraction using wikipedia," in Proceedings of the 48th annual meeting of the association for computational linguistics. Association for Computational Linguistics, 2010, pp. 118–127.

[12] A. Fader, S. Soderland, and O. Etzioni, "Identifying relations for open information extraction," in Proceedings of the conference on empirical methods in natural language processing. Association for Computational Linguistics, 2011, pp. 1535–1545.

[13] M. Schmitz, R. Bart, S. Soderland, O. Etzioni et al., "Open language learning for information extraction," in Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning. Association for Computational Linguistics, 2012, pp. 523–534.

[14] L. Del Corro and R. Gemulla, "Clausie: clause-based open information extraction," in Proceedings of the 22nd international conference on World Wide Web. ACM, 2013, pp. 355–366.

[15] L. Cui, F. Wei, and M. Zhou, "Neural open information extraction," arXiv preprint arXiv:1805.04270, 2018.

[16] Y. Lin, S. Shen, Z. Liu, H. Luan, and M. Sun, "Neural relation extraction with selective attention over instances," in Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), vol. 1, 2016, pp. 2124–2133.

[17] S. Kumar, "A survey of deep learning methods for relation extraction," arXiv preprint arXiv:1705.03645, 2017.

[18] R. Alfred and et al., "Ontology-based query expansion for supporting information retrieval in agriculture," in The 8th International Conference on Knowledge Management in Organizations. Springer, 2014, pp. 299–311.

[19] M. Song, I.-Y. Song, X. Hu, and R. B. Allen, "Integration of association rules and ontologies for semantic query expansion," Data & Knowledge Engineering, vol. 63, no. 1, 2007, pp. 63–75.

[20] M. de Boer, K. Schutte, and W. Kraaij, "Knowledge based query expansion in complex multimedia event detection," Multimedia Tools and Applications, vol. 75, no. 15, 2016, pp. 9025–9043.

[21] M. H. de Boer and et al., "Query interpretation–an application of semiotics in image retrieval," International Journal on Advances in Software, vol. 3 4, 2015, pp. 435–449.

[22] M. H. De Boer, Y.-J. Lu, H. Zhang, K. Schutte, C.-W. Ngo, and W. Kraaij, "Semantic reasoning in zero example video event retrieval," ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM), vol. 13, no. 4, 2017, p. 60.

[23] İ. Pembeci, "Using word embeddings for ontology enrichment," International Journal of Intelligent Systems and Applications in Engineering, vol. 4, no. 3, 2016, pp. 49–56.

[24] G. Wohlgenannt and F. Minic, "Using word2vec to build a simple ontology learning system," in International Semantic Web Conference (Posters & Demos), 2016.

[25] B. Biebow, S. Szulman, and A. J. Clément, "Terminae: A linguistics-based tool for the building of a domain ontology," in Int. Conf. on Knowledge Engineering and Knowledge Management. Springer, 1999, pp. 49–66.

[26] P. Buitelaar, D. Olejnik, and M. Sintek, "A protégé plug-in for ontology extraction from text based on linguistic analysis," in European Semantic Web Symposium. Springer, 2004, pp. 31–44.

[27] T. Declerck, "A set of tools for integrating linguistic and non-linguistic information," in Proceedings of SAAKM (ECAI Workshop), 2002.

[28] P. Cimiano and J. Völker, "text2onto," in Int. Conf. on Appl. of Nat. Lang. to Inf. Sys. Springer, 2005, pp. 227–238.

[29] H. Cunningham, D. Maynard, K. Bontcheva, and V. Tablan, "Gate: an architecture for development of robust hlt applications," in Proceedings of the 40th annual meeting on association for computational linguistics. Association for Computational Linguistics, 2002, pp. 168–175.

[30] A. Zouaq, "An overview of shallow and deep natural language processing for ontology learning," in Ontology learning and knowledge discovery using the web: Challenges and recent advances. IGI Global, 2011, pp. 16–37.

[31] X. Jiang and A.-H. Tan, "Crctol: A semantic-based domain ontology learning system," Journal of the American Society for Information Science and Technology, vol. 61, no. 1, 2010, pp. 150–168.

[32] Y.-B. Kang, P. D. Haghighi, and F. Burstein, "Cfinder: An intelligent key concept finder from text for ontology development," Expert Systems with Applications, vol. 41, no. 9, 2014, pp. 4494–4504.

[33] H. Poon and P. Domingos, "Unsupervised ontology induction from text," in Proceedings of the 48th annual meeting of the Association for Computational Linguistics. Association for Computational Linguistics, 2010, pp. 296–305.

[34] A. Gangemi, V. Presutti, D. Reforgiato Recupero, A. G. Nuzzolese, F. Draicchio, and M. Mongiovì, "Semantic web machine reading with fred," Semantic Web, vol. 8, no. 6, 2017, pp. 873–893.

[35] I. Tiddi, N. B. Mustapha, Y. Vanrompay, and M.-A. Aufaure, "Ontology learning from open linked data and web snippets," in OTM Confederated International Conferences" On the Move to Meaningful Internet Systems". Springer, 2012, pp. 434–443.

[36] R. Bendaoud, A. M. R. Hacene, Y. Toussaint, B. Delecroix, and A. Napoli, "Text-based ontology construction using relational concept analysis," in International Workshop on Ontology Dynamics-IWOD 2007, 2007.

[37] S. Mittal, A. Joshi, T. Finin et al., "Thinking, fast and slow: Combining vector spaces and knowledge graphs," arXiv, no. arXiv: 1708.03310, 2017.

[38] M. Schlichtkrull, T. N. Kipf, P. Bloem, R. van den Berg, I. Titov, and M. Welling, "Modeling relational data with graph convolutional networks," in European Semantic Web Conference. Springer, 2018, pp. 593–607.

[39] J. Brank, M. Grobelnik, and D. Mladenić, "A survey of ontology evaluation techniques," 2005.

[40] H. Hlomani and D. Stacey, "Approaches, methods, metrics, measures, and subjectivity in ontology evaluation: A survey," Semantic Web Journal, vol. 1, no. 5, 2014, pp. 1–11.

[41] P. Spyns, "Evalexon: Assessing triples m ined from texts," STAR, vol. 9, 2005, p. 09.

[42] H. Hlomani and D. A. Stacey, "Contributing evidence to data-driven ontology evaluation workflow ontologies perspective," in 5th International Conference on Knowledge Engineering and Ontology Development, KEOD 2013, 2013, pp. 207–213.

[43] L. Ouyang, B. Zou, M. Qu, and C. Zhang, "A method of ontology evaluation based on coverage, cohesion and coupling," in Fuzzy Systems and Knowledge Discovery (FSKD), 2011 Eighth International Conference on, vol. 4. IEEE, 2011, pp. 2451–2455.

[44] C. Brewster, H. Alani, S. Dasmahapatra, and Y. Wilks, "Data driven ontology evaluation," 2004.

[45] M. Poveda-Villalón, A. Gómez-Pérez, and M. C. Suárez-Figueroa, "Oops!(ontology pitfall scanner!): An on-line tool for ontology evaluation," International Journal on Semantic Web and Information Systems (IJSWIS), vol. 10, no. 2, 2014, pp. 7–34.

[46] A. Burton-Jones, V. C. Storey, V. Sugumaran, and P. Ahluwalia, "A semiotic metrics suite for assessing the quality of ontologies," Data & Knowledge Engineering, vol. 55, no. 1, 2005, pp. 84–102.

[47] A. Gangemi and V. Presutti, "Ontology design patterns," in Handbook on ontologies. Springer, 2009, pp. 221–243.

[48] A. Lozano-Tello and A. Gómez-Pérez, "Ontometric: A method to choose the appropriate ontology," Journal of Database Management (JDM), vol. 15, no. 2, 2004, pp. 1–18.

[49] M. McDaniel, V. C. Storey, and V. Sugumaran, "Assessing the quality of domain ontologies: Metrics and an automated ranking system," Data & Knowledge Engineering, vol. 115, 2018, pp. 32–47.

[50] E. Pianta and S. Tonelli, "Kx: A flexible system for keyphrase extraction," in Proceedings of the 5th international workshop on semantic evaluation. Association for Computational Linguistics, 2010, pp. 170–173.

[51] S. Verberne, M. Sappelli, D. Hiemstra, and W. Kraaij, "Evaluation and analysis of term scoring methods for term extraction," Information Retrieval Journal, vol. 19, no. 5, 2016, pp. 510–545.

[52] M. A. Hearst, "Automatic acquisition of hyponyms from large text corpora," in Proceedings of the 14th conference on Computational linguistics-Volume 2. Association for Computational Linguistics, 1992, pp. 539–545.

[53] Y. Matsuo and M. Ishizuka, "Keyword extraction from a single document using word co-occurrence statistical information," International Journal on Artificial Intelligence Tools, vol. 13, no. 01, 2004, pp. 157–169.

[54] "Countvectorizer," https://scikit-learn.org/stable/modules/generated/sklearn.feature_extraction.text.CountVectorizer.html, accessed: 2019-07-25.

[55] G. Angeli, M. J. J. Premkumar, and C. D. Manning, "Leveraging linguistic structure for open domain information extraction," in Proc. of 53 ACL and 7th Int. Joint Conf. on NLP (Vol 1: Long Papers), vol. 1, 2015, pp. 344–354.

[56] M.-C. De Marneffe, B. MacCartney, and C. D. Manning, "Generating typed dependency parses from phrase structure parses," 2006.

[57] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," in Adv. in neural information processing systems, 2013, pp. 3111–3119.

[58] G. A. Miller, "Wordnet: a lexical database for english," Communications of the ACM, vol. 38, no. 11, 1995, pp. 39–41.

[59] R. Speer and C. Havasi, "Representing general relational knowledge in conceptnet 5." in LREC, 2012, pp. 3679–3686.

Validation of a Failure Cause Searching and Solution Finding Algorithm for Failures in Production

Based on Complaints of a Company in the Field of Stamping and Metal Forming

Marius Heinrichsmeyer
University of Wuppertal
Product Safety and Quality
Wuppertal, Germany
E-mail: heinrichsmeyer@uni-wuppertal.de

Nadine Schlueter
University of Wuppertal
Product Safety and Quality
Wuppertal, Germany
E-mail: schluete@uni-wuppertal.de

Hendrik Dransfeld
University of Wuppertal
Product Safety and Quality
Wuppertal, Germany
E-mail: hendrik.dransfeld@gmx.de

Fynn Koesling
University of Wuppertal
Product Safety and Quality
Wuppertal, Germany
E-mail: fynn.koesling-hk@uni-wuppertal.de

Abstract—**The increasing complexity of new products, production systems as well as customer requirements, bring out huge challenges for companies. Especially regarding the complaint management process, it is necessary to ask how these challenges can still be managed and controlled by individual employees. Current approaches and software systems help to minimize the effect of these challenges but they also reach their limits because of the complexity. For this reason, the research group "Product Safety and Quality" has developed an algorithm to search failure cause and to find a solution to support the complaint management process in identifying failures in the production system and eliminating them. This algorithm was initially based on a conceptual model and was subsequently implemented based on VBA in Excel. The result was the first prototype of this algorithm. To evaluate the algorithm in terms of possible weak points and potentials for improvement, the prototype was validated in the industry in the field of stamping and metal forming. The results of this evaluation are reported in this paper and lessons learned are summarized in the conclusion.**

Keywords-Complexity; Systems; Failure; Algorithm; Complaint; Solution.

I. INTRODUCTION

Due to the Industry 4.0 changes, companies in Germany face ongoing and new challenges: The increasing complexity of products and processes, huge investments, and the lack of skilled workers are just a few that can be listed here [1], [2]. Companies are also confronted with a variety of new but also very individual requirements, which have to be fulfilled in order to maintain competitiveness in the market [2]. In addition, there is the trend that customers want more and more individual, newer and also fascinating products.

However, this trend is critical for companies when customers face failures and complain about the product [3], [4]. In this case, it is important to react very quickly, to recognize the cause of the unfulfilled requirement, eliminate it and thus to satisfy the customer again. But do companies even have the opportunity or the skill to react that quickly?

With the focus on an increasing flow of information in recent years, it becomes apparent that companies are usually unable to handle this amount of information with conventional approaches. This can also be observed in complaint management, where the use of methods such as the 8D-Report and also a catalog of software systems such as RM Babtec, CWA Smart-Process or CASQ-it RUF help to reduce the processing effort and the use of important resources (e.g. time, personnel and costs). The increasing complexity and the increasing relationships between the system elements, on the other hand, are not taken into account [5]. Moreover, many companies have no suitable complaint management process, so that the valuable potential of complaints, e.g., the improvement of production or product system and thus the increase of quality, remains completely unused. Due to this restriction, the worst-case scenario is that the complaint management process is increasingly seen as a burden and not as an opportunity [6], [7]. As a result, further potentials of complaints, including a customer-oriented product development or increased customer satisfaction, will be missed.

To counteract this problem, the research group Product Safety and Quality Engineering is currently developing a targeted failure cause searching and solution finding algorithm for production, based on the fundamental research project "FusLa" (Failure cause searching and solution finding algorithm – Funding indicator: SCHL 2225 / 1-1). With this algorithm, it will be possible to make the potential of available complaint information more useful. The FusLa also has to deal with the complexity of production systems, by choosing a suitable model approach for socio-technical systems. This should not only promote the effectiveness of the complaint processing but also increase the attractiveness for the employees. This is necessary to quickly identify and eliminate failure causes in production in order to reduce or even prevent defective products or wastage. Currently, only the prototype

of the FusLa has been programmed within the project. Nevertheless, the validation is absolutely necessary in order to examine the conceptual model with regard to its applicability.

For this reason, a company, which has its expertise in the field of stamping and metal forming, was engaged for the validation. To protect the company's anonymity and internal know-how, all company-related information has been anonymized. Furthermore, only the results of the validation (running through all four phases) and the question of whether FusLa can currently be used in industry are presented in this paper. [1], [8], [9] prove a scientific gap regarding FusLa after analyzing state of the art and research projects in the field of complaint information probing, prioritization of complaints, failure cause localization in the production or solution finding for failure causes. In order to ensure the transparency of the validation results, Section II gives an overview of current approaches in science and industry. Section III introduces the conceptual model that led to the development of the FusLa prototype. After that, Section IV presents the validation in the stamping and metal forming company and the collected results. Finally, Section V presents conclusions based on the results and an outlook on future research projects.

II. STATE OF THE ART

Some approaches have already prevailed in science and industry. The following Section presents approaches, which are associated with information probing, prioritization, localization of failure cause or solution finding. These are briefly discussed below with regard to their applicability in relation to the present problem:

A. Science

The presented approaches primarily consider the information probing:

The IGF project [10] initially classifies information in a complaint into five categories. Subsequently, the information is analyzed and a first failure image is generated. However, this approach relies on the manual exploration of complaint information. The approach presented in [11] uses text mining algorithms to automatically assess the quality of 8D reports using some metrics such as readability. Although the information in the complaint is probed for text analysis, the focus is on the assessment of the quality and not on the search for a root cause or finding a solution.

The DFG project in [12] initially collects the complaint information uniformly before analyzing and using it. Although the information is eventually used for product development, it does not focus on returning the information to the production.

In the project in [13], a sensor was developed to probe complaint information from online forums. The basis for this was formed by various products and services, but use for the purpose of failure cause localization and solution finding in production was not considered.

The last project associated with information probing, which should be also mentioned, is the project „Learning Failure Management (LeaF)" [14]. In this project, failure data is to be recorded and structured uniformly in order to improve failure management by means of data mining. However, this project is still in its initial phase and therefore cannot be evaluated.

The first of the two publications presented below considers the prioritization of complaints and the second one, the failure cause localization.

In [15] a company-specific approach is described, which prioritizes the complaint in a multi-dimensional way. In the evaluation of failure image, different evaluation dimensions are compared with different failure images and a priority value is determined. However, this approach must be done manually.

The project „DSy" [16] semi-automatically determines the associated failure cause based on the characteristics of a modeled subsystem. However, it only considers embedded systems.

Finally, those approaches have presented that deal with the solution finding. The last contribution of this Section includes all of the presented topics.

The MTQM method proposed in [17] aims to optimize assembly processes based on the previous failure, reliability and cost impact analyzes. However, only assembly processes and prospective actions are considered and no optimization potentials due to actual causes of failure.

A particularly important tool for the analysis of failure-stopping processes is developed in [18]. The quality-related key figures which are developed, among other things, are used for targeting and together with further information to derive recommendations for action.

For complaint analysis, the approach in [19] uses a decision tree, data mining, and the Six Sigma methodology in the framework of DMAIC to propose an approach and tools at each stage. The approach was tested in the gastronomy sector with a positive result. However, it is uncertain how the approach could be customized to complaints or other industries.

B. Industry

Next to the scientific approaches listed under Section II.A, there are also practice-oriented approaches that deal with complaint management. One of these is the 8D report, which is well known but has already been discussed extensively in [1]. The considered software systems usually require manual operation and are based on the knowledge of individuals. For example, it has been shown, because of the extensivity of the information that can be probed by software systems, most of the information must be manually entered into the software.

Even accessing existing databases does not completely solve this problem because the systems are too complex. Similarly, the quantitative, multi-dimensional prioritization of complaints by a software system is not possible. Also, the localization of the failure cause must be done manually as well as the elimination of these.

The approach in [20] uses a smartphone app to allow customers to submit complaints, display failure location, ID, date, time and the complaint reason via a Google Maps interface. Nevertheless, the information collected is too superficial to determine the cause of the failure and furthermore the focus is not on products.

The method presented in [21] is intended to detect quality problems by means of machine learning at an early stage and thus to prevent complaints. Although the approach has great potential, it requires a large number of complaints and focuses on early detection while not describing the solution finding process.

For readout, structuring, and evaluating of complaint documents, [22] uses text and data mining. Accordingly, the documents are examined for similar content to derive patterns of customer complaints. Nevertheless, the approach does not consider the determination of critical organizational areas, failure causes, and solutions.

Finally, the approach according to [23] should be mentioned, in which text mining is also used to analyze customer complaints. However, the focus here is on identifying customer requirements in order to use them for product development. The Outcome-Driven Innovation (ODI) method is used to collect this data and then forward it to the appropriate experts. As before, the large amount of complaints needed is an obstacle and also the focus is not on the localization of failure cause and solution finding.

It has been shown that despite many approaches, which are dealing with complaints and preparation of information through different methods such as Text Mining, there is so far no approach that considers failure cause localization and solution finding. Furthermore, there are no interfaces between text mining and company-specific information systems in order to make the existing knowledge available for the system.

III. FAILURE CAUSE SEARCHING AND SOLUTION FINDING ALGORITHM (FUSLA)

The prototype of the failure cause searching and solution finding algorithm was programmed based on a conceptual model. The conceptual model, which is summarized in Figure 1, is based on extensive analysis of literature, the evaluation of 13 different software systems that are currently on the market, and the adaptation of available approaches of artificial intelligence to deal with large amounts of data. In order to develop the model, requirements based on the advantages and disadvantages of the respective research projects and software systems were derived. The baseline is a conceptual model, which considers the phases of production as well as usage. It divides the algorithm into a total of four main phases [1]. The production phase includes all processes that are needed to produce the corresponding product. The usage phase, on the other hand, focuses on the application of the product direct by the customer. The four main phases are "information probing", "prioritization", "failure cause localization", and "solution finding". In the first phase (information probing), relevant complaint information is probed from the arising complaint text during the usage phase. Based on that, quantitative values are calculated in the second phase (prioritization) to define a mostly objectively priority of the complaint. The priority is calculated using both the complaint information probed from the first phase and information from the information systems of the organization. With the knowledge of the most relevant complaint, the third phase (failure cause localization) is initiated. It serves to localize the existing failure cause within a production system during the production phase. In the final phase (solution finding), the localized failure cause is finally eliminated [1]. In order to make the individual phases and their processes more transparent, these are explained in more detail in the following Sections. This paper deliberately talks about information and not data. Unlike many research projects, the present paper defines information (e.g., a specific temperature specification) as linked data (e.g., numerous measured values of temperature). So it is talked about complaint information and not complaint data.
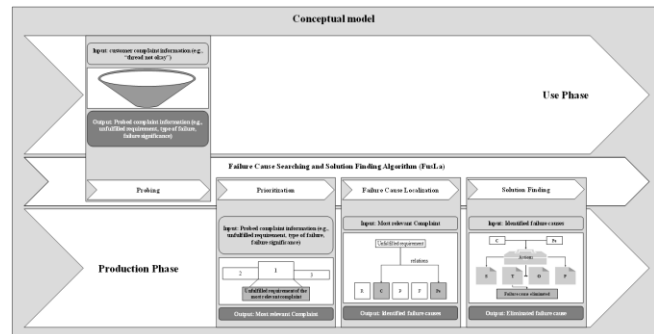

Figure 1. Conceptual model of the Failure Cause Searching and Solution Finding Algorithm

A. Information Probing

Information probing is the starting point for effective and goal-oriented failure cause searching and solution finding. In this phase, the complaint text will be analyzed for relevant complaint information. Current approaches, as shown in [1], using manual evaluation of complaint texts, which increases the need for resources in comparison with an automated evaluation. However, the conceptual model of FusLa provides a different approach for the information probing. Automated processes pursue the goal of collecting relevant complaint information more effectively. But how can such an automated procedure be realized? The answer is hidden in the connection between existing information systems of the company and the analysis of the complaint text. Unlike the fact that an employee, who is responsible for the handling of complaints, is not able to permanently monitor all information systems of the company, the FusLa has exactly such an interface. This means that the FusLa decomposes the complaint text into individual text modules, compares those with the existing information systems in the company (e.g., customer or order system) and collects all necessary information (e.g., customer or order numbers) for further processing. In order to achieve this comparison, however, a structure was developed according to differentiate relevant and less relevant complaint information. In addition, rules to specify the procedure for the FusLa have been defined. The structure includes a total of six different types of information, including, information about the complaining organization or information about the order of the complained product. These types are further divided into individual information modules, such as the name or the identification number of the complaining organization.

Figure 2.   Structure of the Information Probing, including all types of information as well as the information modules

This means that information types, such as "contact information", act as a group and the information modules only reflect the respectively relevant information, for example, the "name of the contact person". The corresponding information types and information modules were derived on the basis of the results of the literature analysis and software evaluation [1].

In addition to the structure, some rules were assigned to the algorithm. These rules are defined as for/if loops and determine the decisions and boundaries of the algorithm. A total of six different loops were defined, which are used, for example, to recognize the unfulfilled requirement or to recognize customer information. Each of these loops makes it possible to filter different information modules from the complaint text or the existing information systems, in order to finally establish a solid basis for the further evaluation of the complaint.

To increase the comprehensibility of the loops, the first loop for "identifying the order number" is explained as an example. The presentation of other loops is discussed in detail in [24] and is therefore not part of this paper. As already mentioned, the algorithm can probe the relevant complaint information by comparing the information contained in the complaint text with the information saved in the information systems of the company. In the first loop, a technique of Natural Language Processing (NLP), the so-called Tokenization, is used initially. Above all, a distinction between Values and Strings is made here possible. These tokens are then compared with the order numbers in the organization's order system. If the algorithm identifies a match, it extracts the corresponding order number and identifies related information, including the customer or the delivered product and automatically filters it. If the probing is unsuccessful, the algorithm will start the next loop.

For reference, the defined information types and information modules are summarized in Figure 2.

B. Prioritization

The prioritization presents the second phase of the FusLa. This phase is absolutely necessary in order to recognize the complaints, which are most critical for a company, at an early stage.

Current approaches from both science and industry make use of subjective prioritization methods or categorizations/rank orders [1]. Since one-dimensional prioritization, which is based on subjective impressions may lead to an incorrect complaint priority, it is inappropriate to use such methods. To avoid that, the conceptual model of the FusLa provides a multi-dimensional prioritization, which is mostly objective and thus less prone to misjudgment.

In order to be able to realize this form of prioritization, a total of nine different dimensions is derived, including "Classification of the Customer (D1)" or the "Failure History (D8)". All of these dimensions include a dimension value and a weighting value. Since a detailed explanation of the dimensions exceeds the scope of this paper, these dimensions are briefly summarized in Table I. However, in order not to reduce the transparency by the short presentation of the individual dimensions, a detailed presentation of the dimensions is set out in [25], including the individual values and weightings which are ultimately required for the calculation of the priority.

TABLE I.   DIMENSIONS OF PRIORITIZATION

No.	Dimension	Description
1	Customer Classification	Customers are ranked for their relevance to the organization. Based on this ABC classification, a quantitative classification is derived.
2	Date information	The due date and the date of submission of the complaint are essential for the calculation of the remaining work required to deal with the complaint. In addition, the urgency also influences the priority.

No.	Dimension	Description
3	Amount of complaint products	The extent of an unfulfilled requirement, which is calculated by the proportion of products complaint in total, is recorded in the 3rd dimension.
4	Repetitions	If the unfulfilled requirement occurs in several batches of one product, it can be count as a repetition. This parameter shows how often the unfulfilled requirement has been recorded and increases the priority on repeated occurrence.
5	Failure Type	It can be distinguished between very different failures, for example, dimensional failure or document failure. Depending on the severity of the failure for the customer, measured by the type of failure, the priority will be influenced.
6	Failure Meaning	Similar to the failure type, a distinction is also made between the meaning of an unfulfilled requirement. Based on the impact, for example, on the customer or merely the product itself, it decides the importance of failure and thus the priority of a complaint.
7	Product Sales	The product sales divided by the total sales of an organization show the importance of a product for an organization. Accordingly, this ratio is also included in the priority.
8	Failure History	The number of single failures of a product in terms of the total number of failures for all products over the entire production period shows the proportion of failures compared to all other product failures. The larger the share the more critical the complaint.
9	Amount of Costs	Similar to the failure history, the proportion of the individual costs by complaints of a product in terms of the total costs of all products over the entire production period can be evaluated. This shows, which resources are spent on handling the complaint.

Based on the presented dimensions, quantitative values are calculated and included in a final priority for a complaint. In order to be able to guarantee the comparability and compatibility of the individual dimension values, these are additionally normalized to a range of 1-10. A slightly wider, but also narrower interval can be selected. For this purpose, only the formula needs to be adjusted. It should be considered, however, that the distinction between values decreases at closer intervals. In this case, a range of 1-10 is sufficient. The formula (1) derived from [26] is used for this purpose:

$$NDW_{ij} = 9 \cdot \left(\frac{DW_{ij} - DW_{min, \, ij}}{DW_{max, \, ij} - DW_{min, \, ij}} \right) + 1 \qquad (1)$$

DW_{ij} = Dimension value of the dimension j of the complaint i
$DW_{min, \, ij}$ = Minimum of the dimension values of the dimension j of the complaint i
$DW_{max, \, ij}$ = Maximum of the dimension values of the dimension j of the complaint i
NDW_{ij} = Normalized dimension value of the dimension j of the complaint i

The normalized dimension values are used in a further step to derive a weighting of the respective dimension. This step is necessary to mostly objectively assess the importance of the dimension to the organization itself. In order to do this, all dimension values of all the so far completed complaints are summed up to form an average value. This ensures that, on the one hand, all complaints that have been processed are included in the weighting, and on the other hand, the weighting automatically adapts with new complaints. Once the weighting of the respective dimensions has been determined, the priority of the actual complaint is calculated in the last step. Furthermore, the following formula is used.

$$P_i = \sum_{j=1, \; i=1}^{n} NDW_{ij} \times G_{ij} \qquad (2)$$

G_{ij} = Weighting value of the dimension j of the complaint i
NDW_{ij} = Normalized dimension value of the dimension j of the complaint i

After prioritization, the third phase of the failure cause searching and solution finding algorithm can be initiated with the most relevant complaint. What this phase looks like is explained in the following Section.

C. Failure Cause Localization

During the failure cause localization phase, the FusLa should identify all possible causes of failures within the production system. An evaluation of current approaches with regard to their ability to localize the cause of failures, it turns out that they rely on a subjective assessment by a single employee or team [1]. Despite the experience or expertise, the employee responsible for handling the complaint is not able to observe the enormous complexity of a production system, e.g. with all its components, processes or persons. This is not necessarily due to the employee, but simply to the variety of system elements and their interaction. Particularly in the case of changing production systems or newly hired employees, a subjective search for failure causes is nearly impossible.

To counteract this problem, the FusLa was developed based on the so-called Enhanced Demand Compliant Design (eDeCoDe) approach by Winzer [27] and Nicklas [28], to the modeling socio-technical systems. It uses five standardized views (requirements, components, functions, processes, and person) to classify system elements. Using different matrices, including Design Structure or Domain Mapping Matrices, the interrelations between the individual system elements are recorded and displayed transparently with a notation of 1 (relation) and 0 (no relation). This principle is shown in Figure 3 as an example.
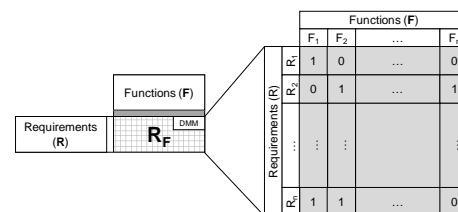

Figure 3.   Example of a Domain Mapping Matrix

The merging of all matrices, in the form of a multi-domain matrix, results in a model of the production system, which can be mapped in the form of a multi-domain graph as shown in Figure 4 as an example.
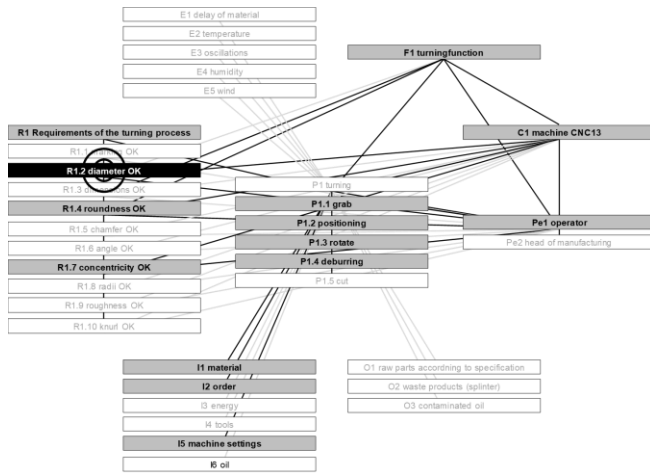

Figure 4.   Localization example based on [1]

Based on this model, it is now possible to evaluate the interrelations between the individual system elements. The FusLa examines which notation, whether 0 or 1, exists between a system element (e.g. the positioning process) and the unfulfilled requirement (in this case the diameter). If this notation is present, the algorithm identifies the corresponding system elements as a possible cause of the failure. This is extremely helpful, especially with regard to the available complaints. In this way, the traceability can be ensured comprehensibly and transparently.

This means that it is possible to retrace the unfulfilled requirement from the complaint to the interrelated system elements, e.g., components (machines) or processes (turning process). Without this possibility, the failure cause localization is also very random and less goal-oriented.

D. Solution Finding

With the last phase of the FusLa, failure causes which are highly probable can be eliminated by appropriate measures. This is essential to avoid the repetition of complaints regarding the same requirement. In order to define effective actions, it makes sense to identify the most likely failure causes. In order to estimate the probability of failure causes, different categories of failure cause information were formed. These categories, as summarized in Table II, are intended to provide a semi-quantitative estimate of probability.

TABLE II.   FAILURE CAUSE INFORMATION CATEGORIES

Failure Cause (System Element)	Failure Cause Information
Requirement	Definition date of the requirement
	Release date of the requirement
	Drawing index for the requirement (Adjustment delivery and drawing)
Component	The failure rate of the component

Failure Cause (System Element)	Failure Cause Information
	Availability of the component
	Maintenance date of the component
Process	Standardization status of the process
	Release date of the process
	Environmental conditions of the process
	The correctness of the input of the process
	The capability of the process
Person	Technical competence (process-dependent) of the person
	Methodical competence (process-dependent) of the person
	Personal competence (process-dependent) of the person

It can be seen that different information on failure causes is analyzed depending on the system element. Among other things, for a component-related failure cause (e.g., a machine), quantitative values such as failure rate or availability may be used for the probability estimation. Thus, for example, in the case of a high failure rate, a high probability of causing the unfulfilled requirement can be assumed. However, in the case of personal failure causes (e.g., operators), process-dependent competencies are identified and compared with the person's competences to determine whether the person was fit to perform the process. If deviations are detected with regard to their competences, the person seems to be a very probable failure cause. With this approach, it is possible to identify the most likely failure causes.

In order to be able to eliminate those, it is necessary to derive appropriate actions. This is achieved through the application of the "STOP principle", a classification, which was able to establish itself in the field of safety engineering [29]. The STOP principle provides four different forms of actions, including substitutive (S), technical (T), organizational (O) and personal (P) actions. It is intended to start with personal actions because these are usually quick to implement and do not require a huge amount of resources. If these actions are not sufficient, organizational, then technical and ultimately substitutive actions have to be derived.

To illustrate what such actions may look like, Table III shows an excerpt of actions for person-related failure causes.

TABLE III.   ACTIONS FOR PERSON RELATED FAILURE CAUSE

Failure Cause: Person	
Substitutional Actions (S)	
Action 1	Job Rotation of the Person
Technical Actions (T)	
Action 1	Implement an IT system for knowledge transfer and training
Action 2	Provide knowledge base or wikis for best practice solutions
Organizational Actions (O)	

Failure Cause: Person	
Action 1	Improve education and training opportunities
Action 2	Prepare standardized process instructions for processes
Action 3	Training for newly recruited persons, who act as a substitute or for persons with part-time employment
Action 4	Integrate competencies into the quality management system
Action 5	Plan and conduct conversations and coaching sessions
Action 6	Promote a working atmosphere for employees through additional offers
Action 7	Incorporate personnel support more strongly in the corporate philosophy
Action 8	Selection of suitable persons for the execution of the process
Personal Actions (P)	
Action 1	Regular sensitization of persons
Action 2	Instruction and advice to the persons
Action 3	Supervision of persons by supervisors

Table III illustrates that very different measures are proposed by the algorithm depending on the type of action, i.e., S, T, O or P. However, the user of the algorithm is still left to decide, which action he wants to select and implement. The decision provided by the user is then documented for subsequent complaints and taken into account for similar complaints.

In order to investigate the applicability of the presented algorithm, it was validated in a company in the field of stamping and metal forming. The results are shown in the next section.

IV. VALIDATION OF THE FAILURE CAUSE SEARCHING AND SOLUTION FINDING ALGORITHM

The validation of the prototype of FusLa is a very important step to highlight the improvement potentials as well as weaknesses to initiate the continuous improvement process. For the validation, a company in the field of stamping and metal forming was engaged. This company provided insights into complaint management and application of the algorithm through several complaint examples. Again, it is pointed out that in order to preserve the anonymity of the company and to protect its internal know-how, all company-related information has been anonymized, so that the following chapter only shows the validation results.

It should also be noted that the implementation in Visual Basic for Applications (VBA) does not serve to create a software system based on the conceptual model. Otherwise, it serves only to prove the applicability of the conceptual model of the algorithm by its specific application. Due to the fact that MS Office is used as a documentation basis in many companies, it makes sense to simplify the interface through the deliberate use of VBA and not, for example, Python or C++.

In order to be able to present the validation results in a transparent manner, both the company and the application example are presented briefly.

A. Presentation of the company for stamping and forming technology

The company, which agreed to validate the FusLa, has its expertise in the field of stamping and metal forming. Among other things, this field includes the production of a wide variety of stamped, bent and deep-drawn parts. Due to the broad product portfolio, different industries are covered. As a basis for the validation, the company provided us with complaints about various available products. Due to the extent of all validation results, this paper deals only with the complaint of the KSGD product. Nevertheless, it can be summarized that the results of further validations correspond to those of the KSGD product, but in particular, the massive influence of the quality of the complaints on the assessment by FusLa can be emphasized.

Figure 5. Complaint text of the product KSGD

Another complaint was validated analogously for comparison possibilities. It can be seen that only the abbreviations of the selected product are mentioned. This serves to protect the internal know-how and possible anonymity reduction by tracing the products back to the company.

B. Preparation of company-related information

In order to be validated, it is necessary to collect all useful company-related information. These are saved in very different information systems. In order not to have to establish every interface to each information system, the information systems were modeled in the form of an excel

sheet. However, the principle of access to information systems is not affected. In total, three different information systems are replicated.

These include the customer system, the order system, and the product system. Based on the replication of the information systems, a model of the production system was created using the eDeCoDe approach. Altogether 74 requirements, 8 functions, 29 processes with 22 inputs and 11 outputs, 9 components, and 10 involved persons are elicited. Furthermore, inspections of documents (e.g., technical drawings, inspection plans) are analyzed and some discussions are held with process owners.

C. Application of the Failure Cause Searching and Solution Finding Algorithm

With the help of this information basis, the actual application of the FusLa could be carried out.

1) Information Probing

The application of the FusLa is initiated by the first phase, the information probing. The provided complaint text is transferred to the FusLa and the information probing is started as shown in Figure 6.

Figure 6.   Information Probing_1 for the complaints of the product KSGD

From Figure 6, it can be seen that the algorithm declared different information as relevant and translated them into the appropriate fields of the information probing surface. By accessing the customer system of the company, the algorithm also succeeded in capturing all other information, for example, with regard to the customer's address or its classification. It also turns out that the algorithm is able to probe the correct information for the further handling of the complaint. This is a mandatory requirement for goal-oriented and precise failure cause searching and solution finding.

In the next step, the algorithm should probe all relevant order and frame information regarding the claimed product. Among other things, the product name, product number or the order number itself are listed here. The first weaknesses of the prototype of the algorithm were revealed during this step. Due to the fact that the complaint text contained no product number, order number or even batch, the algorithm did not recognize, which product-specific delivery is concerned. The weaknesses are presented in Figure 7. This shows that it makes sense to establish the possibility of recognizing the product name. In this case, the algorithm would have recognized the name KSGD and probs all relevant product information. For this reason, the information probing is extended by another loop, which also checks the product name in complaint text.

Figure 7.   Information Probing_2 for the complaints of the product KSGD

Another weakness was shown regarding the due date of the complaint. There was no deadline in the complaint text, so the algorithm could not track how many days the organization would have left for handling the complaint. For this reason, another loop has been added to the algorithm. This allows the algorithm to access the contractual provisions between the customer and the organization with regard to the processing time interval of complaints. This information is used by the algorithm to calculate a determination of the due date. If this process is unsuccessful, precisely defined strings are also implemented, which allow the algorithm to probe out time limits from the complaint text.

In the final step of information probing, the algorithm should probe all failure and failure scope information. That means it should recognize the unfulfilled requirement and assign a failure type as well as meaning to it. The result of the validation shows that the algorithm recognizes that there is a failure (e.g., flatness out of tolerance), but it does not select this failure correctly. Within the complaint text, an exact indication of the unfulfilled requirement was missing. So the user had to choose between the different requirements of flatness and thus received a direct estimate of the type and meaning of the failure. This process is presented in Figure 8.

Figure 8. Information Probing_3 for the complaints of the product KSGD

How this problem can be solved is not yet defined. In the further phases of the project, however, the aim is to probe out the unfulfilled requirement via the customer's failure description in the complaint text.

Another problem was the exploration of the costs. In discussion with the company, it was found that the customer usually makes no statement about the costs and mostly only assigns an estimate of the affected products. In this case, no costs and only an estimation of 1-2 affected products were included in the complaint text. This is not enough for automated evaluation. However, since, according to the company, this information only occurs in a few complaints, it should be questioned whether it is actually relevant to the handling of the complaint or whether it should only be collected at the end of the handling process. Currently, an optional exploration of the costs is considered, in which the user decides whether and which costs should be extracted. However, this is not sufficient if automation of the process is the goal.

2) Prioritization

After all steps of the information probing have been carried out, the second phase of the FusLa, the so-called prioritization, is to be examined. The basis for the prioritization is the complaint information generated in the information probing phase. In order to check whether the algorithm can make a reasonable prioritization for the corresponding complaints, the complaint of the KSGD was compared to a second complaint about a product called SHD. Similar to KSGD, product SHD is also from the automotive sector. Since the comparison limited to two complaints, the dimension values are in the range of 1, 5 or 10 and the weighting values in the range of 5 and 5.5. Nevertheless, this is sufficient to investigate how meaningful the results are. The evaluation of the prioritization is illustrated in Figure 9.

Figure 9. Prioritization for the complaints of the product KSGD

From Figure 9, it can be seen that the algorithm classifies the complaint with a high priority of 316.5. The evaluation of prioritization initially shows that normalization is successful for a range of 1-10 for each dimension. It also turns out that each of the dimension values and weighting values is plausible on the basis of the previously probed information. Also, the priority value for this complaint is comprehensible and correct. Thus it could be shown that the prioritization was successful from a mathematical point of view.

Nevertheless, it must be questioned whether the priority value also reflects the reality of the company. Through a discussion with the employees of the complaints management department, it could be confirmed that this

complaint was very relevant and furthermore, properly prioritized. The result is an effective prioritization of the complaint. In order to work out whether the localization of failure causes is also meaningful, this phase is examined in the next step.

3) Failure Cause Localization

The failure cause localization is determined both on the complaint information recorded in the information probing and also on the production system model developed in the preparation of the information basis. In order to be able to record the actual cause of the unfulfilled requirement, the relations of the requirement to the other system elements were examined. The result was a selection of possible failure causes, which is illustrated in Figure 10 based on the eDeCoDe views. It should be noted that the functional view of the realization is represented by the component view and is furthermore, not listed again.

Figure 10. Failure Cause Localization for the complaints of the product KSGD

It can be seen that the algorithm could assign different failure causes. Among other things, four different components (machines/tools) and three processes were identified. Nevertheless, a variety of improvement potentials emerged when locating the cause of the failure. On the one hand, it became clear through a discussion with the complaint manager that the failure causes are indeed plausible, but that they do not cover all possible causes. The company also wanted to display the system elements, which are not only used to realize the requirement but are also expected to influence the requirement within the previous or following process, such as processes of hardening or surface treatment. This would be recommended especially

for very case-specific complaints. This was realized by the eDeCoDe model of the production system, which now also considers indirect relations between the system elements.

A key weakness of the algorithm was illustrated by the estimation of the semi-quantitative probability over the defined failure cause information of Table II. It turned out that much of the information, as shown in Figure 11, was not collected at all and furthermore, cannot be used as a basis for valuation.

Figure 11. Failure Cause information (explicit Component C1: Ef) for the complaints of the product KSGD

In addition, it became apparent that a comparison between actual and required input is hardly possible without a specification by the customer. Competencies could be used for the persons, but these were also not complete. Furthermore, an adaptation of the probability estimation in the algorithm must be developed, in order to provide results even if there is an insufficient information basis. It would make sense to provide an interface to existing systems, for example, CAQ systems or smart machines, which allow access to real-time-based data and information. The documentation of such data/information has to be also continuous and complete.

4) Solution finding

In the last phase of the algorithm, actions to eliminate the cause of the failure should be derived. To implement this phase, actions based on the STOP principle is used. Figure 12 shows how these tools were presented by the algorithm.

Figure 12. Actions (explicit Component C1: Ef) for the complaints of the product KSGD

It can be seen that the algorithm provides different actions to the user. These must be manually selected by the user to ensure the best possible combination of actions. The result of the validation showed that the algorithm is able to present helpful guidelines. The possibility of selecting predefined actions saves resources like time. Likewise, the user can also enter individual alternative solutions. With the help of all these possibilities, each of the recognized failure causes could be eliminated. Nevertheless, this process is currently still manual. The idea is to implement this process via an AI. It can be considered only in further research projects.

V. CONCLUSION AND FUTURE WORK

The validation of the FusLa based on complaints from the company in the field of stamping and metal forming showed both the possibilities and the weaknesses of FusLa. Beginning with information probing, it turned out that while the algorithm is able to distinguish relevant information from less relevant information, the FusLa still has problems probing it. Although the algorithm was able to record the contact person of the customer, it was not possible to probe the order information because of the insufficient validation of the complaint text. The next phase of prioritization showed that the algorithm has the ability to face and prioritize complaints. Through collaboration with the company, the meaningfulness of the prioritization value for the complaint could be confirmed. However, the third phase of the algorithm still poses major problems. On the one hand, although the algorithm recognized failure causes, these did not include the system elements, which not only realizes the requirement but also influences it. In terms of probability estimation, the algorithm reached its limits due to a lack of information. With the last phase, the solution finding, it was possible to derive suitable actions based on the STOP principle. These were quite capable of eliminating the most likely failure causes.

The quintessence shows that the algorithm has great potential, but also it is still in the stage of a prototype.

In order to improve the algorithm, the following main research areas are needed. First of all, a study should be done on how the quality of different complaint texts affects the information probing. This would highlight the need for standardization of complaint texts or more capable natural language parser. Furthermore, it is necessary to investigate how the probability of failure causes can be derived on the basis of an incomplete failure cause information. Only in this way it is possible to differentiate between more probable and less probable failure causes. Lastly, the user-friendliness of the algorithm for people in the industry should be investigated. Since the industry is the most important user of such an algorithm, it must achieve acceptance.

REFERENCES

[1] M. Heinrichsmeyer, N. Schlüter, A. Ansari, "Algorithm for Dealing with Complaints Data from the Use Phase," in Proceedings ICONS 2019, vol. 1, Sendra C., S. IARIA Publisher Valencia, Spain, pp. 1–6, 2019.

[2] M. Neuhold, S. Bley, J. Gudat, "*Industrie 4.0 Status Quo und Perspektiven*," [Online]. Available from: https://www.ey.com/Publication/vwLUAssets/ey-industrie-4-0-status-quo-und-perspektiven-dezember-2018/$FILE/ey-industrie-4-0-status-quo-und-perspektiven-dezember-2018.pdf, 07.05.2019.

[3] R. Early, "Guide to Quality Management Systems for the Food Industry," Springer Science+Business Media LLC, New York, 1995.

[4] A. K. Rai, "Customer relationship management," PHI Learning, vol. 2, New Delhi, 2013.

[5] D. Fallmann, "Datenberge erfolgreich bezwingen," in Wissensmanagement, Lehnert, O. Büro für Medien Oliver Lehnert e.K. Neusäß, pp. 41–45, 2012.

[6] S. Cook, "Complaint management excellence," Kogan Page, London, 2012.

[7] M. Zairi, "Managing customer dissatisfaction through effective complaints management systems," The TQM Magazine, vol. 12, pp. 331–337, 2000.

[8] M. Heinrichsmeyer, "Algorithmus zum Umgang mit Reklamationsinformationen aus der Nutzungsphase," in Umgang mit Anforderungen in agilen Organisationen, Schlüter, N.; Reiche, M. Shaker Publisher Aachen, pp. 113–128, 2018.

[9] M. Heinrichsmeyer, N. Schlüter, A. Ansari, "Algorithm based handling of complaints data from the usage phase," in The Proceedings of 2019 International Conference on Quality, Reliability, Risk, Maintenance, and Safety Engineering, Huang, H.-Z.; Beer, M.; Lim, J.-H. Zhangjiajie, Hunan, China, 2019.

[10] R. Schmitt and A. Linder, "*IREKS – Ein ganzheitlicher Reklamationsprozess – Unternehmensinterne Strukturen zur effektiven Analyse, Bearbeitung und Nutzung von Kundenreklamationen,*" [Online]. Available from: http://www.ireks.rwth-aachen.de/de/811309b639b3aec9c1257daf00476b0d/141215_Kurzbericht_IREKS.pdf, 26.07.2019.

[11] M. Marchenko, G. Ullmann, B.-A. Behrens, L. Overmeyer, "Exzellentes Reklamationsmanagement," Zeitschrift für wirtschaftlichen Fabrikbetrieb (ZWF), vol. 12, pp. 1102–1107, 2010.

[12] T. Hellebrandt, I. Heine, R. H. Schmitt, "Knowledge management framework for complaint knowledge transfer to product development," Procedia Manufacturing, pp. 173–180, 2018.

[13] R. H. Schmitt, A. I. Metzmacher, V. Heinrichs, B. Falk, "Customer Language Processing," in 12th International Workshop on Semantic and Social Media Adaptation and Personalization, Bieliková, M.; Šimko, M. IEEE Piscataway, NJ, pp. 32–33, 2017.

[14] M. Lindemann, "Digitales Fehlermanagement," Qualität und Zuverlässigkeit, vol. 08, pp. 8, 2018.

[15] R. Schmitt and A. Linder, "*Beschwerden zielgerichtet priorisieren,*" [Online]. Available from: http://www.wzlforum.rwth-aachen.de/__C12571ED003C17E6.nsf/html/a995b44ece6b2815c1257af000394446.html/$FILE/MQ_2012_11_Reklamationen_WZL.pdf, 22.09.2018.

[16] Görschwin Fey, "*DSy - Debugging Eingebetteter Systeme,*" [Online]. Available from: http://gepris.dfg.de/gepris/projekt/165955509/ergebnisse, 22.09.2018.

[17] R. Refflinghaus, T. Trostmann, C. Kern, J. Palomo, "Menschlichen Fehlern auf die Schliche kommen," QZ - Qualität und Zuverlässigkeit, vol. 4, pp. 34–37, 2019.

[18] R. Exner, "*Projekt PROFAP,*" [Online]. Available from: http://www.profap.rwth-aachen.de/de/veroeffentlichungen.html, 15.05.2019.

[19] Y.-H. Hsiao, L.-F. Chen, Y. L. Choy, C.-T. Su, "A novel framework for customer complaint management," The Service Industries Journal, vol. 13-14, pp. 675–698, 2016.

[20] M. Greedharry, V. Seewoogobin, Sahib-Kaudeer, Nuzhah Gooda, "A Smart Mobile Application for Complaints in Mauritius," in Information Systems Design and Intelligent Applications, vol. 2, Satapathy, S. Chandra; Bhateja, V.; Somanah, R.; Yang, X.-S.; Senkerik, R. Springer Singapore, pp. 345–356, 2019.

[21] S. Aris-Brosou, J. Kim, L. Li, H. Liu, "Predicting the Reasons of Customer Complaints," JMIR medical informatics, vol. 2, pp. e34, 2018.

[22] E. A. Stoica and E. K. Özyirmidokuz, "Mining Customer Feedback Documents," International Journal of Knowledge Engineering-IACSIT (IJKE), vol. 1, pp. 68–71, 2015.

[23] J. Joung, K. Jung, S. Ko, K. Kim, "Customer Complaints Analysis Using Text Mining and Outcome-Driven Innovation Method for Market-Oriented Product Development," Sustainability, vol. 1, pp. 40, 2019.

[24] M. Heinrichsmeyer, N. Schlüter, A. Ansari, "Algorithmus zur automatisierten Abfrage relevanter Informationen aus Kundenreklamationen," in Bericht zur GQW-Jahrestagung in Aachen, Schmitt, R., pp. Status: Accepted, 2019.

[25] M. Heinrichsmeyer, I. M. Lemke, N. Schlüter, "Development of an automated prioritization procedure for complaints," in The Proceedings of 22nd QMOD/ICQSS conference (Quality Management and Organisational Development/ an International Conference on Quality and Service Sciences), Dahlgaard, J.; Dahlgaard Park, S. Mi Krakow, Poland, 2019.

[26] R. Alt, "Statistik," Linde Publisher, vol. 2, Wien, 2013.

[27] P. Winzer, "Generic Systems Engineering," Springer Vieweg Publisher, vol. 2, Berlin, Heidelberg, 2016.

[28] J.-P. G. Nicklas, "Ansatz für ein modellbasiertes Anforderungsmanagement für Unternehmensnetzwerke," Shaker Publisher, vol. 1, Aachen, 2016.

[29] M. Egger and O. Razum, "Public Health," De Gruyter, vol. 2, Berlin, 2014.

Discovering Hotspots with Photographic Location and Altitude

from Geo-tagged Photographs

Masaharu Hirota
Faculty of Informatics
Okayama University of Science
Okayama-shi, Okayama
Email: hirota@mis.ous.ac.jp

Jhih-Yu Lin
Graduate school of System Design
Tokyo Metropolitan University
Hino-shi, Tokyo
Email: lin-jhihyu@ed.tmu.ac.jp

Masaki Endo
Division of Core Manufacturing
Polytechnic University
Kodaira-shi, Tokyo
Email: endou@uitec.ac.jp

Hiroshi Ishikawa
Graduate school of System Design
Tokyo Metropolitan University
Hino-shi, Tokyo
Email: ishikawa-hiroshi@tmu.ac.jp

Abstract—A hotspot is an interesting place where many people go sightseeing. A place where many photographs have been taken (which we call a hotspot) might be an interesting place for many people to visit. Analyzing such places is important to promote industries such as those related to tourism. To identify hotspots, most existing research applies a grid-based or density-based clustering algorithm, such as density-based spatial clustering of applications with noise (DBSCAN) or mean shift. When applying such methods to hotspot detection, the features used for clustering are latitude and longitude. Therefore, the identified hotspots are visualized in a two-dimensional space. However, large areas, landmarks, and buildings may include elevated hotspots or multiple hotspots with different altitudes, which cannot be distinguished by latitude and longitude. Therefore, in this research, we propose methods for identifying hotspots based on altitude, in addition to latitude and longitude, and visualizing these hotspots in a three-dimensional space. We propose two types of method, based on density-based and grid-based clustering, that use these features. The first method is one that improves ST-DBSCAN, which clusters data based on spatial and time features. The other method is an extension of general grid-based clustering using these features. As an example application, we classified the identified hotspots as shooting spots, observation spots, areas of interest, and others. We demonstrate our approach by identifying hotspots in a three-dimensional space using photographs obtained from Flickr, and discuss the usefulness of detecting hotspots using altitude in addition to latitude and longitude.

Keywords–area of interest; density-based clustering; grid-based clustering; photograph location; clusteing.

I. INTRODUCTION

A considerably shorter pre-version of this paper has already been published in [1].

Owing to the increasing popularity of mobile devices, such as digital cameras and smartphones, numerous photographs have been uploaded to photo-sharing web services, such as Flickr [2], Instagram [3], and Google Photos [4]. In addition, these digital devices have recently been equipped with Global Positioning System (GPS) sensors. Thus, many photographs are annotated with latitude and longitude information, which shows the place where the photograph was taken. If many people take photographs in the same area, this may be an area of interest. As described in this paper, we define such areas as hotspots. The identified hotspots are used to analyze urban areas [5] and tourist behavior [6].

Many methods have been proposed to identify hotspots [7], [8], [9], [10], [11]. Most existing research on detecting hotspots is based on a density-based or grid-based clustering method, such as density-based spatial clustering of applications with noise (DBSCAN) [12] and mean shift [13]. The studies that apply these methods use latitude and longitude as features to identify high-density areas as clusters, and the detected clusters are then defined as hotspots. However, clusters obtained by such a method, only using latitude and longitude, do not consider altitude. Therefore, there are some cases where multiple hotspots, at different altitudes, are identified as one hotspot. For example, in a sightseeing location such as the Eiffel Tower, the latitude and longitude for the observatory and the area around the tower are almost the same, but there are several hotspots with different altitudes. Even if the altitude is different, because these latitudes and longitudes are almost equal, it is difficult to distinguish between these hotspots.

In our previous research [1], we proposed a method for detecting hotspots, using ST-DBSCAN [14], which deals with the time when the photograph was taken, in addition to latitude and longitude. When we apply ST-DBSCAN in this paper, we use altitude instead of time to detect hotspots, thereby considering the height of the hotspot. In this paper, as an alternative approach, we propose a method using grid-based clustering. Previous research has been conducted on grid-based clustering for detecting hotspots in two-dimensional space. In this paper, we extend the method to three-dimensional space. In addition, this paper compares the two types of proposed method, by the visualization results and execution time.

The remainder of this paper is organized as follows. Section II describes the difference of extracted hotspots on 2D or 3D. Section III describes work related to this topic. Section IV presents our two types of proposed method for detecting hotspots based on altitude. Section V presents several examples of visualization results, and experiment on execution time by our proposed methods. Section VII concludes the paper with a discussion of the results and future work.
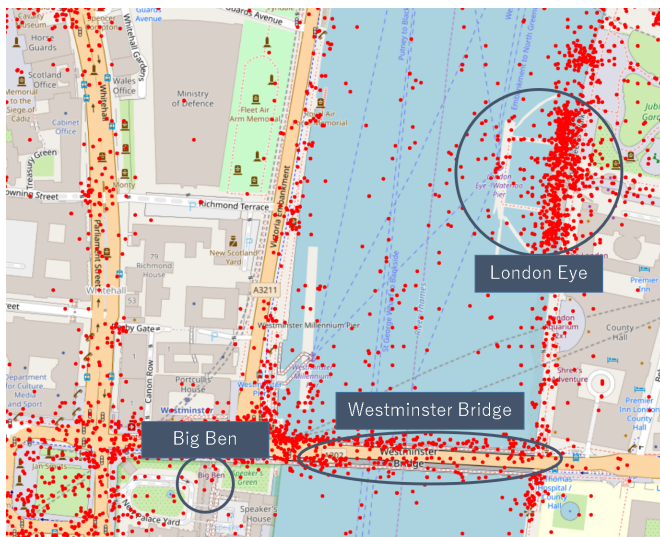
Figure 1. Visualization of photograph location on 2D.

TABLE I. Famous landmarks in London.

Name	Latitude	Longitude
Big Ben	51.500729	−0.124625
London Eye	51.503324	−0.119543
Westminster Bridge	51.500942	-0.121874

Figure 2. Clustering result by DBSCAN.

Figure 3. Visualization of photograph location on 3D.

II. Extracting hotspots on 2D and 3D

In this section, we discuss the difference between extracted hotspots using methods on 2D and 3D.

Figure 1 visualizes the photographs obtained from Flickr. According to the latitude and longitude, we superimposed these photographs on a map of OpenStreetMap [15]. These photographs were taken in London. There are several famous buildings in this area, which are listed in Table I. The red points in the figure show locations of photographs (obtained from Flickr) that were taken around Big Ben, the London Eye, and Westminster Bridge. The number of photographs is 5,000, which we randomly selected from the obtained photographs. There are some areas around the three tourist attractions that have a high density of points.

We applied DBSCAN to these data and detected 23 clusters. Figure 2 shows the clustering result. The points in this figure show the photograph locations that were classified as belonging to clusters. Each color in the figure represents one cluster (the colors are only used to distinguish visually between the clusters). Figure 3 shows the visualization results of the same photograph location data as in Figure 1, but using latitude, longitude, and altitude in a three-dimensional space. The blue points in this figure show the photograph locations. This figure clearly shows the diversity of the altitudes of the photograph locations. The red points in Figure 1 around the London Eye have a high density. Figure 3 shows that there are actually two high-density groups of locations in the area of the London Eye: at the top of the wheel and at ground level. We believe that these two groups should be identified as distinct clusters. However, in Figure 2, these photograph locations are regarded as a single cluster. If hotspots are identified using latitude and longitude, distinction between those hotspots is difficult, but using altitude information makes it possible to distinguish them.
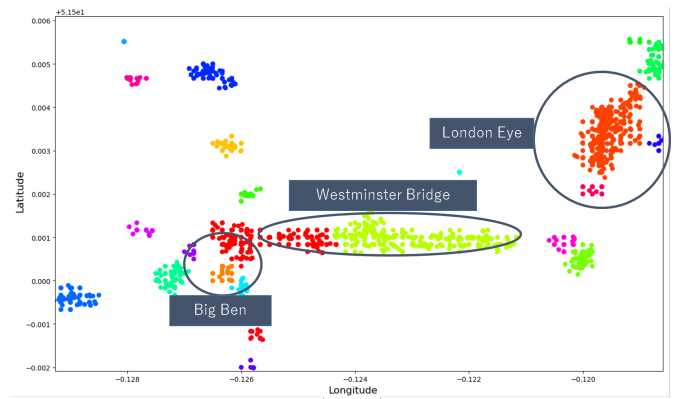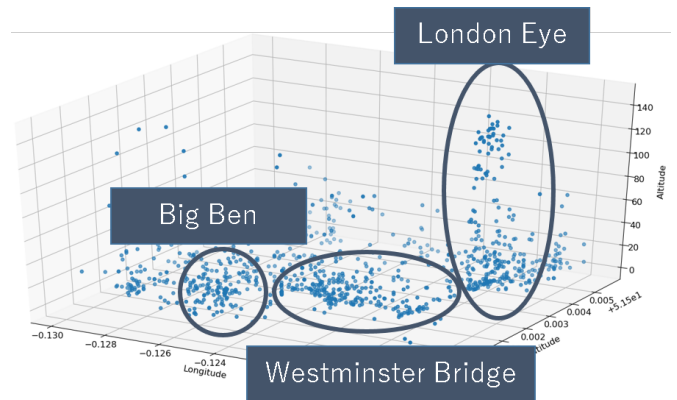
Common methods for detecting hotspots such as DBSCAN and mean shift treat the distance between two points as one dimensions using latitude and longitude. We believe that these methods are inappropriate for clustering with three-dimensional metadata. In this research, we propose approaches that considers not only the area of a hotspot, represented by latitude and longitude, but also the height of the hotspot, by adding the altitude.

Additionally, hotspots can be classified into three types: an area of interest, a shooting spot, and an observation spot [16], [17]. Areas of interest include tourist attractions (for example, the Colosseum or the Statue of Liberty). In such areas, many photographs have been taken inside the site or at a nearby location. However, when people take a photograph of such an attraction, they will take it at a place that is at some distance from the attraction itself. Such places are also identified as hotspots, and are defined as shooting spots. Finally, observation spots are hotspots for photographing the surroundings of the hotspot. In this research, we classify hotspots, detected considering the altitude in addition to latitude and longitude, into three classes by considering multiple information sources, such as the direction of photography, and we then visualize the results.

III. Related Work

In this section, we discuss some work related to our study, including detecting hotspots and analyzing the detected

hotspots.

A. Detection of hotspots

Various methods have been proposed to detect hotspots in a large dataset using location information. In the case of detecting hotspots from large datasets of photographs annotated with location metadata, two main approaches are used: density-based clustering algorithms, such as DBSCAN [12] and mean shift [13], and grid-based clustering algorithms.

Crandall et al. presented a method to detect hotspots using mean shift based on many photographs annotated with photograph location [18]. Kisilevich et al. proposed P-DBSCAN, an improved version of DBSCAN, for the definition of a reachable point, to detect hotspots using the density of photograph locations [19]. Ankerst et al. proposed a clustering method OPTICS, which is a variant of DBSCAN used to create a cluster using different subspaces extracted from various parameters [20]. Sander et al. proposed GDBSCAN, which extends DBSCAN to enable correspondence to both spatial and non-spatial features [21]. Shi et al. proposed a density-based clustering method to detect places of interest using spatial information and social relationships between users [22]. Chen et al. proposed a clustering method for massive sets of spatial points based on density peaks and connect [23]. Yang et al. proposed a method to identify human mobility hotspots using kernel density estimation to evaluate convergent and dispersive hotspots [10]. Yang et al. proposed an algorithm to detect hotspots of various sizes using self-tuning spectral clustering [8]. Kulkarni et al. proposed a parameter-free method for detecting hotspots from spatiotemporal trajectories without any *a priori* assumptions [9].

Another famous approach to detecting hotspots is the grid-based clustering algorithm. In grid-based clustering, the data space is quantized into a finite number of cells, which are formed by the grid structure. Whether a cell is a cluster is determined by the number of data points included in the cell. The main advantage of grid-based clustering algorithms is a fast processing time: most of the algorithms achieve a time complexity of $O(n)$, where n is the number of data points (compared with, for example, DBSCAN, whose complexity is $O(n \log n)$ using tree algorithm such as a k-dimensional tree [24]) [25]. Moreover, the performance of grid-based clustering depends only on the size of the grid, which is usually much less than the number of data points [26]. Additionally, most grid-based clustering algorithms are easy to parallelize because each cell is independent when the algorithm detects whether the cell is defined as a cluster. Agrawal et al. proposed CLIQUE, to detect clusters within subspaces of the dataset using an *a priori*-like technique [27]. Wang et al. presented STING, which combined grid-based and density-based approaches [28]. Chang et al. proposed the axis-shifted grid-clustering algorithm, which performs a dynamic adjustment to the size of the original cells in the grid and a reduction in the weakness of the borders of cells [29].

The studies outlined above detect hotspots using a density-based clustering method, such as DBSCAN, or a grid-based clustering method, both based on latitude and longitude. However, in some cases, actual hotspots include the concept of height and are distributed in three-dimensional space, rather than a two-dimensional space. In this paper, we propose two new types of approach to detecting and visualizing hotspots using ST-DBSCAN or grid-based clustering, by adding altitude to latitude and longitude.

B. Detecting hotspots based on photograph orientation

As photographs with a photograph orientation have become more commonly available, photograph orientation has been increasingly useful for detecting hotspots.

Photograph orientation is an important information source, because it constitutes information about the user's interests. The photographer shoots a subject of his or her own interest from some location. The direction of the subject is combined with the photograph location. Therefore, hotspots are likely to exist in the directions taken by users.

Lacerda et al. proposed a method for detecting hotspots using photograph orientations [30]. This method calculates intersections between the lines of the orientations of many photographs. The intersections are then clustered using DBSCAN. In addition, Thomee et al. proposed a method for considering the inaccuracies affecting GPS location measurements [31]. Hirota et al. proposed a method for determining the areas of hotspots using the orientation and angle of view of photographs [32]. This method determines the area from the overlaps of pseudo-triangles calculated by photograph orientation, location, and some other metadata.

The above methods focus on photograph orientation to detect areas where the users' interest is concentrated. Therefore, in this paper, our approach classifies the types of detected hotspots based on the photograph orientation and the users' interest.

C. Analysis of detected hotspots

Some researchers have studied approaches to analyze hotspots obtained from large photograph datasets annotated with various metadata, such as location.

The method to detect hotspots is used to find or detect geographical characteristics. Spyrou et al. proposed a method to understand the underlying semantics of detected hotspots using user-generated tags [33]. Omori et al. evaluated georeferenced photographs annotated with user-generated tags related to coastlines, to show the actual coastline [34]. Hu et al. proposed a method to understand urban areas from detected hotspots using user-generated tags, to choose preferable photographs based on the image similarity between photographs of the hotspot [35]. Chen et al. proposed a framework to detect boundaries of hotspots from geotagged data, and used them to construct spatiotemporal profiles of areas [36]. Zhu et al. proposed a method for analysis of emotions of detected hotspots [37].

There are also some methods to detect the relationship between a hotspot and another hotspot, such as the relationship between photograph subjects and shooting spots. Shirai et al. proposed a method to detect a hotspot using DBSCAN and to calculate the relation between hotspots [38], [16]. To discover a wide area of interest, this approach infers the relation between hotspots based on the photograph location and orientation. Hirota et al. proposed a method to detect and visualize various relationships between hotspots using photograph orientation and social tagging [17]. The above researchers have extracted various relationships from detected hotspots.

The areas of interest detected by our proposed method represent areas in which many people took photographs. We

apply these studies to detected hotspots and expect to be able to analyze the results in more detail.

IV. PROPOSED METHOD

In this section, we describe our two types of proposed method for detecting hotspots considering the altitude, in addition to the latitude and longitude, of photographs. We also describe our proposed method for classifying the hotspots into three types: area of interest, shooting spot, and observation spot, using photograph location and orientation.

A. Extracting hotspots with altitude using ST-DBSCAN

Here, we explain why we adopt ST-DBSCAN to detect hotspots with altitude, in addition to latitude and longitude. In most of the previous research, DBSCAN has been used for detecting hotspots. Until now, latitude and longitude have been used as features for calculating the distance between two points. Because we now need to consider altitude to detect hotspots, we infer that neither DBSCAN nor mean shift is an appropriate method for this purpose. This is because Eps, which is the parameter of those methods for evaluating the distance between two points, is a one-dimensional threshold. As previously described, there are hotspots with different altitudes but almost equal latitude and longitude. Therefore, although those methods are appropriate for using latitude and longitude as one feature for evaluating the distance between two points, it is not appropriate to add altitude to the feature. As a result, altitude should be regarded as a different feature from latitude and longitude, and we adopt ST-DBSCAN to achieve this.

Moreover, when applying DBSCAN, there is an approach to evaluate distances between photographs using the three features of latitude, longitude, and altitude with one distance function, and detect the cluster with one Eps threshold. However, this is an inappropriate approach for our purposes. The values of latitude and longitude have similar characteristics, but altitude is different from them. Therefore, in this paper, rather than considering latitude, longitude, and altitude as a single feature, we detect hotspots using latitude and longitude as one feature and altitude as the other feature.

ST-DBSCAN is one of the improved methods of DBSCAN that considers time in addition to the spatial feature of latitude and longitude. ST-DBSCAN has three parameters—$Eps1$, $Eps2$, and $MinP$—where $Eps1$ is a threshold of distance between the spatial features of two photographs, $Eps2$ is a threshold of distance between other features, such as the difference between the times when two photographs were taken, and $MinP$ is a threshold of the number of photographs included in the cluster.

Here, we describe the procedure of ST-DBSCAN.

- The method extracts the core data points such that the number of neighborhood data points within $Eps1$ and $Eps2$ is greater than $MinP$.
- The method evaluates the distance between the core data points and others.
- If the distance is less than $Eps1$ and $Eps2$, these data points are connected. The method defines connected data as a cluster.
- The data points for which the number of neighborhood data points (within $Eps1$ and $Eps2$) is less than $MinP$ are defined as noise data and not included in any cluster.

Figure 4 shows an overview of detecting hotspots using ST-DBSCAN. In this figure, the red points are regarded as core data and detected as a cluster, whereas the blue points are regarded as noise data

In this research, we apply ST-DBSCAN, with $Eps1$ as latitude and longitude and $Eps2$ as altitude.

In the implementation of ST-DBSCAN, we use a k-dimensional tree [24] to search neighborhood data. Here, because ST-DBSCAN needs two types of distance, this method constructs two k-dimensional trees: one for latitude and longitude and another for altitude.

B. Detecting hotspots with altitude using grid-based clustering

Our second method for detecting hotspots uses the grid-based clustering approach. Figure 5 shows the procedure of grid-based clustering. This method constructs a three-dimensional grid space with latitude, longitude, and altitude. We map photographs to the grid and count the number of photographs. We extract voxels that contain many photographs and connect the extracted adjacent voxels. The connected voxels are regarded as hotspots.

First, we map the photographs that have a photograph location to the grid. Using the assigned grid coordinate, we count the number of photographs. Photograph p_i is mapped to coordinates (x_i, y_i, z_i), as shown below.

$$z_i = M_{alt} - \frac{(p_i^{alt} - Alt_{min}) * M_{alt}}{Alt_{max} - Alt_{min}} \quad (1)$$

$$y_i = M_{lat} - \frac{(p_i^{lat} - Lat_{min}) * M_{lat}}{Lat_{max} - Lat_{min}} \quad (2)$$

$$x_i = M_{lng} - \frac{(p_i^{lng} - Lng_{min}) * M_{lng}}{Lng_{max} - Lng_{min}} \quad (3)$$

Here, Alt_{max}, Lat_{max}, and Lng_{max} denote the maximum values of altitude, latitude, and longitude, respectively; Alt_{min}, Lat_{min}, and Lng_{min} denote the corresponding minimum values. M_{alt}, M_{lat}, and M_{lng} are the height, length, and width of the grid. (This is decided using a parameter m to adjust the number of cells required in this procedure. In this paper, we set these parameters to be the same as the ST-DBSCAN parameters $Eps1$ and $Eps2$.) Consequently, each cell in the obtained grid includes a photograph taken in the range.

Using the obtained grid, we extract the cells for which the number of photographs included in the cell is greater than the threshold $MinP$.

At this stage, having determined whether each individual cell is a hotspot, we connect any extracted hotspots that are in adjacent cells. We calculate the distance $D(c_p, c_q)$ between cells c_p and c_q as the Chebyshev distance, as follows.

$$D(c_p, c_q) = max_i(\|c_{pi} - c_{qi}\|) \quad (4)$$

where c_{pi} and c_{qi} represent the feature value of c_p and c_q, respectively, in the i-th dimension. We connect the two cells if $D(c_p, c_q)$ is 1, which means that those cells are adjacent; otherwise the cells are not adjacent.

Finally, each group of joined cells is defined as a hotspot.

C. Classification of hotspot

In this paper, a hotspot is classified as an area of interest, a shooting spot, or an observation spot (as shown in Figure 6) using the orientation annotations of the photographs included
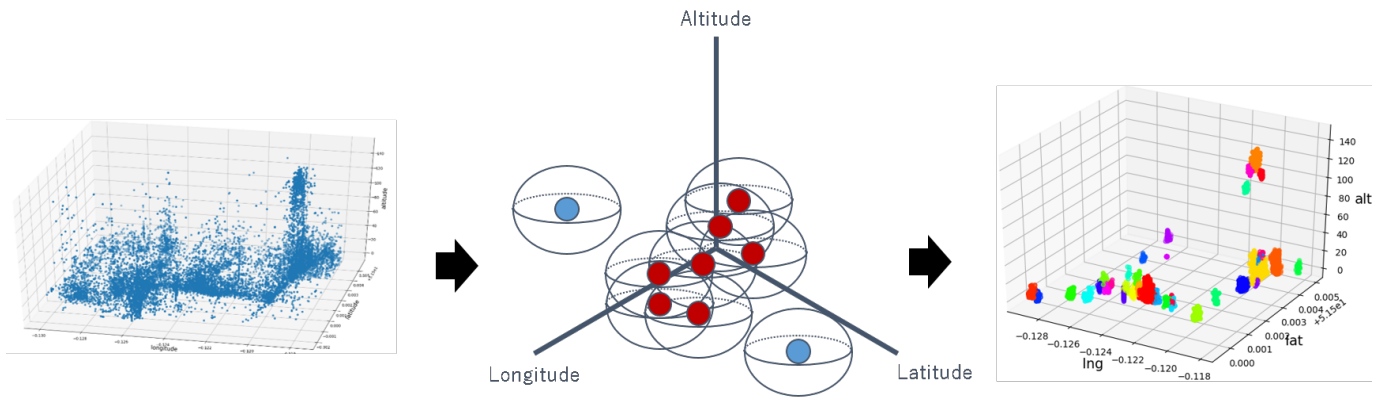
Figure 4. An overview of clustering by ST-DBSCAN.
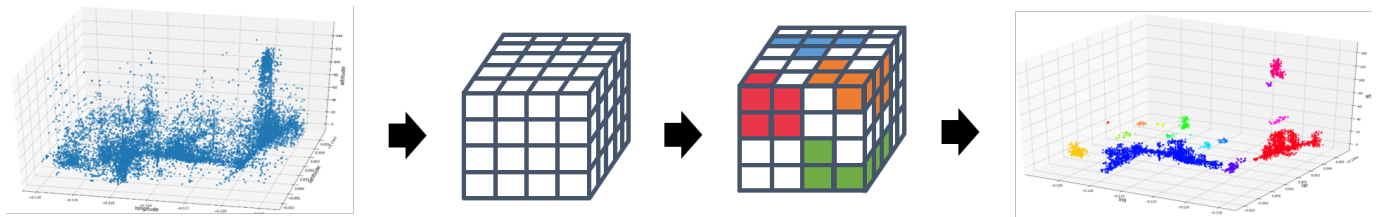
Figure 5. An overview of grid-based clustering.
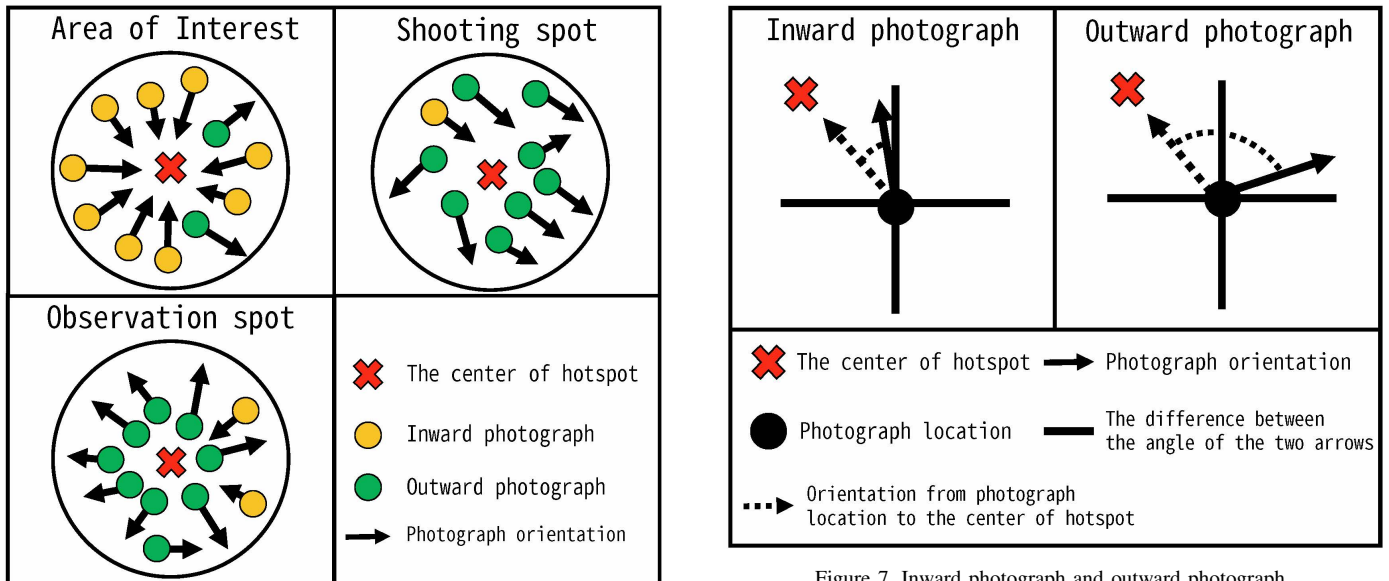
Figure 6. Classification of hotspots.

Figure 7. Inward photograph and outward photograph.

in the detected hotspots. However, the number of photographs with a known photograph orientation (in addition to latitude and longitude) is minuscule compared with the number of photographs with only latitude and longitude. As a result, the classification of hotspots that have few photographs may be difficult. Therefore, we classify hotspots into four groups: areas of interest, shooting spots, observation spots, and others. In this research, we classify hotspots that have less than 10 photographs with a known orientation as "other", and we do not perform the following processes on them.

First, we decide whether each hotspot is a shooting spot. In this case, many photographs are taken with a specific orientation. Therefore, we calculate the bias of the photograph orientation based on its frequency distribution. We divide the value of the photograph orientation by 10 degrees and count the number of photographs in each of the 36 classes. We consider a hotspot to be focused on a specific orientation if the largest class includes at least 15% of the photographs belonging to the hotspot.

In the next step, we classify each remaining hotspot as either an area of interest or an observation spot according

to the photograph orientation. Moreover, this classification is based on the ratio of inward to outward photographs in the hotspot. Figure 7 shows examples of inward and outward photographs. In this research, if the photograph orientation and the orientation to the center of gravity of the hotspot are close, we regard the photograph as an inward photograph; otherwise, we classify it as an outward photograph.

We set the orientation (with the true north as $0°$) of the photograph to θ_i. If the coordinates (latitude and longitude) of the center of gravity of the hotspot are (x_h, y_h) and the coordinates of the shooting position are (x_i, y_i), θ_d is the orientation from the shooting position to the center. We calculate the orientation θ_d in which (x_h, y_h) exists using the following equation:

$$\theta_d = \tan^{-1} \frac{\cos y_i \times \sin(x_h - x_i)}{\cos y_1 \times \sin y_h - \sin y_i \times \cos y_h \times \cos(x_h - x_i)} \tag{5}$$

Next, we classify each photograph in a hotspot as an inward or outward photograph based on the difference between θ_d and θ_i, as follows:

$$\begin{cases} inward & |\theta_i - \theta_d| < \theta \\ outward & \text{otherwise} \end{cases} \tag{6}$$

In this study, we set the threshold for classifying inward and outward photographs as $\theta = 50$. If the number of photographs classified as inward photographs is larger than the number of outward photographs, the hotspot is classified as an area of interest; otherwise, it is classified as an observation spot.

V. EXPERIMENT

This section presents a description of experiments conducted using our proposed method. We present and discuss several examples of detecting hotspots by density-based and grid-based clustering.

A. Dataset

Here, we describe the dataset for the experiment of detecting hotspots. Photographs for the experiments were obtained from Flickr, and included metadata for latitude, longitude, altitude, and orientation. In this paper, we used the exchangeable image file format (Exif) metadata for latitude (GPSLatitude), longitude (GPSLongitude), altitude (GPSAltitude), and orientation (GPSImgDirection).

The dataset included photographs taken in an area of Westminster in London (latitude: 51.5056 – 51.4979; longitude: -0.1178 – -0.1299). The size of this area is about 1×1 km. We obtained photographs taken between January 1, 2011 and May 10, 2016.

To deal with altitude errors, we set a threshold for altitude and removed photographs having an altitude higher or lower than the threshold. In this experiment, we set this parameter based on the height of buildings around the area to be analyzed. In addition, we removed photographs with an altitude of 0 m or less.

Furthermore, we excluded photographs in which the latitude, longitude, and altitude all overlap in the dataset. This might occur as a result of an incorrect GPS position or device configuration. Such points for which there is much inappropriate metadata is excessively evaluated when detecting
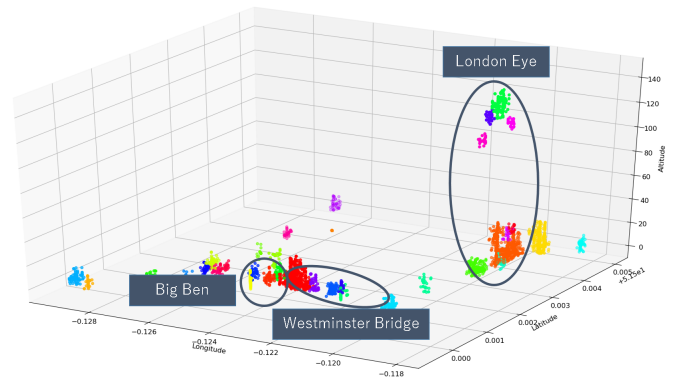
Figure 8. The clustering result in three-dimensional (ST-DBSCAN).
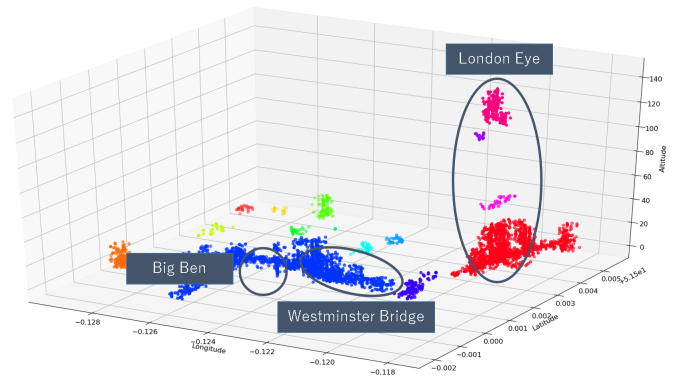
Figure 9. The clustering result in three-dimensional (grid-based clustering).

a hotspot. As a result, the number of photographs used in this experiment was 13,911.

B. Visualization of hotspots

Figure 8 shows the clustering results by ST-DBSCAN, based on the latitude, longitude, and altitude of photographs. The parameters used in ST-DBSCAN and grid-based clustering were $Eps1 = 0.0001$, $Eps2 = 5$, and $MinP = 30$, respectively. Figure 9 shows the clustering results by our proposed grid-based method. The parameters of this method is the same as for ST-DBSCAN. The number of clusters detected by ST-DBSCAN (Figure 8) was 35, and the number detected by grid-based clustering (Figure 9) was 6. In Figures 8 and 9, photograph locations classified as noise are not displayed. In these figures, each color represents a cluster (the colors are only used to distinguish visually between the clusters).

Figure 8 shows that some clusters with different altitudes were detected in areas with almost the same latitude and longitude. In particular, several clusters were detected near an altitude of 130 m, latitude of 51.504, and longitude of -0.120. This is because the highest point of the London Eye is 135 m. Therefore, many people take photographs around there, and the area was detected as a hotspot.

Compared with ST-DBSCAN (Figure 8), the detected clusters were more widespread using grid-based clustering (Figure 9). However, Figures 8 and 9 show that the detected hotspots cover almost the same areas. Instead of evaluating the distance between two photographs by latitude and longitude in DBSCAN, the grid-based clustering detects hotspots by
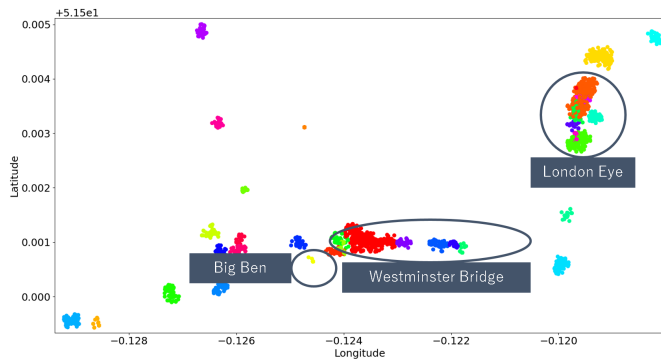
Figure 10. The clustering result in two-dimensional (ST-DBSCAN).
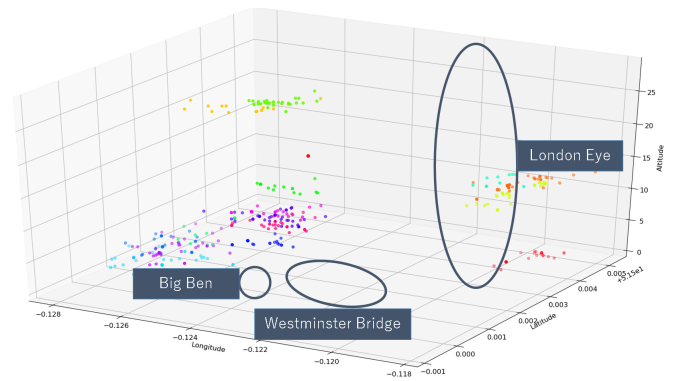
Figure 12. Clustering result by DBSCAN using latitude, longitude and altitude.
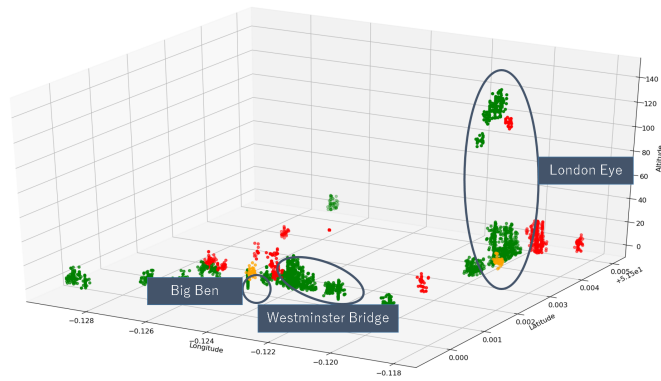
Figure 11. Classification result of hotspot.
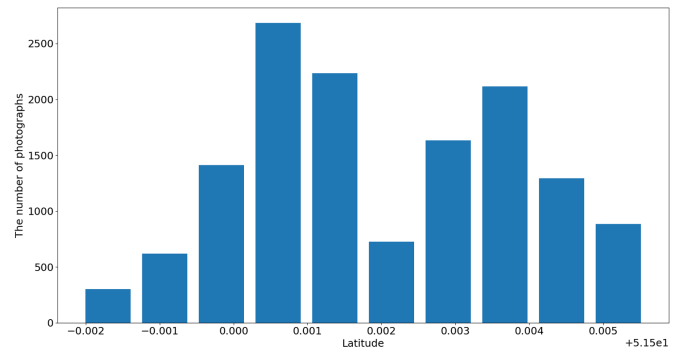
Figure 13. Histogram of Latitude of photographs.

evaluating the relationship between adjacent cells. Therefore, the photograph locations that were classified as multiple clusters by ST-DBSCAN were classified as one cluster by our proposed grid-based clustering method. At this stage, it is not possible to determine which method is better, because quantitative evaluation has not been performed for either method. Therefore, as future work, it is necessary to examine which method can detect hotspots more accurately.

Figure 10 shows a two-dimensional representation of the clustering result by ST-DBSCAN (i.e., the figure shows the clusters in Figure 8 mapped in two dimensions, without altitude). Some clusters are displayed overlapping in multiple areas in this figure. Therefore, in such areas, points with different altitudes should be identified as belonging to distinct hotspots. Naturally, the latitude and longitude of the photographs taken in such areas are almost equal. Unless we detect hotspots by considering the altitude, in addition to latitude and longitude, it is difficult to distinguish between, and correctly detect, these clusters.

Although it may be possible to distinguish these hotspots by clustering with only latitude and longitude in some cases, substantial time and effort would be required to tune the Eps and $MinP$ parameters in DBSCAN. In addition, when photographs annotated with latitude and longitude are used, these metadata often include errors. Therefore, photographs that should belong to different hotspots may erroneously be assigned to the same hotspot. Therefore, in Figures 8 and 10, we show that it is possible to distinguish between hotspots in areas with similar latitude and longitude by considering the altitude, even in such a state.

C. Combination of features

Next, we discuss the advantages of our proposed methods, compared with simply applying DBSCAN and using latitude, longitude, and altitude as one feature. We applied DBSCAN to the London dataset using latitude, longitude, and altitude; Figure 12 shows the clustering results. The parameters of DBSCAN were the same as ST-DBSCAN, except for the threshold $Eps2$ of altitude, and use Euclidean distance to calculate the distance between each pair of data points. The number of detected clusters was 15. The clusters in Figure 12 tend to have a flat shape, because the scale of the features varied greatly.

We show the histograms of latitude, longitude, and altitude in Figures 13, 14, and 15. In this dataset, the high density area is around latitude 51.50001 and longitude -0.120, shown in Figures 13 and 14. The distributions of latitude and longitude depend on areas of high density included in the dataset. However, the distribution and the scale of altitude are very different: altitude varies between about 0 and 140, as shown in Figure 15. This distribution depends on the area being analyzed and the height of the landmarks in that area. In most cases, the distribution of altitude is different from that of latitude and longitude.

As a result, when measuring the distance between data points, altitude becomes a more dominant feature than latitude and longitude. In addition, because the distribution of these features is very different, it is difficult to handle them consistently even if the feature values are normalized.

Figure 14. Histogram of Longitude of photographs.

Figure 16. Execution time of each method.

Figure 15. Histogram of Altitude of photographs.

Therefore, when calculating the distance between each pair of data points in our methods for detecting hotspots, we believe that latitude and longitude should be treated together, and altitude treated separately.
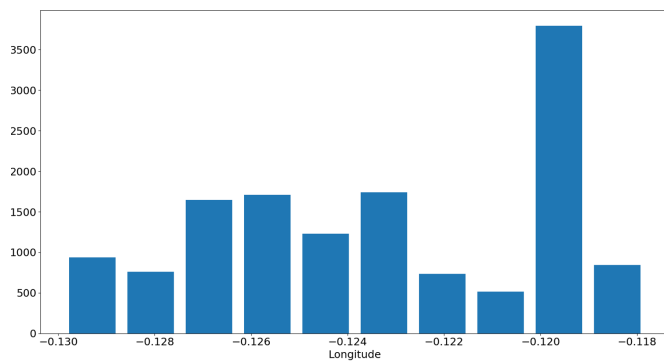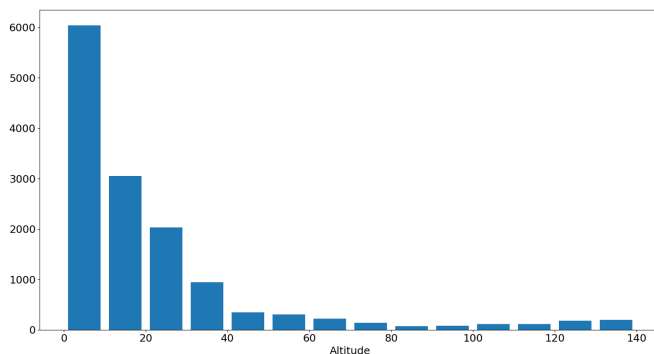
D. Execution time

In this section, we compare the execution time for detecting hotspots by each method. We measured the time of DBSCAN and our proposed ST-DBSCAN and grid-based clustering. The experiment used the dataset of London photographs, as described in Section V-A. We used five datasets with varying numbers of photographs, from 2,000 to 10,000. The parameters of these methods were the same as used in Section V-B.

Figure 16 shows the execution times of the three methods: these are the median times for performing each method ten times. In Figure 16, the grid-based clustering is the fastest of the three methods. This reason is that this method has a low computational complexity $O(n)$ (while the complexity of DBSCAN and ST-DBSCAN is $O(n \log n)$. Therefore, as the number of data points increases, the execution time hardly increases, compared with the other methods.

ST-DBSCAN has a smaller execution time than DBSCAN, even though the two methods use almost the same algorithm. ST-DBSCAN uses the latitude and longitude threshold $Eps1$ and altitude threshold $Eps2$. In contrast, DBSCAN uses only the latitude and longitude threshold $Eps1$. Therefore, compared with DBSCAN, because ST-DBSCAN needs to satisfy two conditions when searching for neighborhoods, fewer photographs are classified as belonging to a cluster. As a result, the execution time of ST-DBSCAN is less, because the number of times the algorithm determines connectivity with

surrounding data is reduced.

From Figure 16, the grid-based clustering method is much faster than ST-DBSCAN. However, because the grid-based method needs to map photographs to cells when detecting hotspots, the detected hotspots are rougher than those of ST-DBSCAN as shown in Figure 9. Therefore, it is desirable to select these approaches properly, considering this result.

VI. VISUALIZATION OF CLASSIFIED HOTSPOTS

In this section, we show the result of classifying hotspots into four types. Figure 11 shows the result of the classification of hotspots which is detected by ST-DBSCAN. In this figure, a green point shows a photograph location in a hotspot classified as an observation spot, a red point is a shooting spot, and an orange point is an area of interest. In Figure 11, many observation spots were detected: for example, the highest location of the London Eye is an observation spot. It seems that people are shooting the view from the top of the Ferris wheel. In addition, there are two clusters of orange points: under the London Eye and around Big Ben. These hotspots should probably be classified as shooting spots because these contents of photographs include the landmarks that is these photographs are shot in the hotspots. The area around latitude 51.502 and longitude −0.121 is detected as a shooting spot because it includes many photographs of Big Ben. It seems that other areas are also classified as shooting spots because they contain many photographs of landmarks, such as the London Eye and Big Ben.

In the above description, we explained the classification results regarding hotspots. At this stage, quantitative analysis of the classification has not been performed. In the results, some hotspots have been misclassified. Therefore, in a future study, there is a need to improve the method for, and evaluation of, the classification of hotspots.

VII. CONCLUSION

In this paper, we proposed two types of method for detecting hotspots using the geographical coordinates of photographs (latitude, longitude and altitude). The first of our proposed methods uses ST-DBSCAN, which is a density-based clustering method. The other is a grid-based clustering method. We visualized the clustering results using those methods based on the metadata of photographs taken in London. We discussed the detection of separate hotspots that may be detected as a single cluster when considering only latitude and longitude. We compared these proposed methods with DBSCAN, which uses latitude, longitude, and altitude as a single feature of a

photograph. As a result, we showed that we can detect hotspots that overlap when we consider only latitude and longitude. In addition, we compared the execution time for detecting hotspots by these methods. Finally, we visualized the clustering results and classified hotspots as areas of interest, shooting spots, and observation spots.

As future work, we aim to compare our approach with clustering methods other than ST-DBSCAN and grid-based clustering. In this paper, these methods have been applied using latitude, longitude, and altitude as features, but it has not yet been revealed to be superior quantitatively to other clustering methods, such as DBSCAN and mean shift. Moreover, our method of grid-based clustering is still naive; we will improve this method to enable it to detect hotspots faster. For example, we can consider the characteristics of features to speed up the decision of whether a cell is a hotspot. We performed classification of hotspots but have not quantitatively evaluated the result yet. The results obtained suggest that further improvements in our proposed classification method are necessary.

References

[1] M. Hirota, M. Endo, and I. Hiroshi, "A proposal for discovering hotspots using 3d coordinates from geo-tagged photographs," in Proceedings of The Eleventh International Conference on Advanced Geographic Information ystems, Applications, and Services, ser. GEOProcessing 2019, Feb 2019, pp. 59–62, ISBN:978-1-61208-687-3, URL:http://www.thinkmind.org/index.php?view=article&articleid=geoprocessing_2019_4_10_30046.

[2] "Flickr," URL: https://www.flickr.com [accessed: 2019-07-15].

[3] "Instagram," URL: https://www.instagram.com/ [accessed: 2019-07-15].

[4] "Google photos," URL: https://photos.google.com/ [accessed: 2019-07-15].

[5] Z. Xia, H. Li, Y. Chen, and W. Liao, "Identify and delimitate urban hotspot areas using a network-based spatiotemporal field clustering method," ISPRS International Journal of Geo-Information, vol. 8, no. 8, 2019.

[6] Y. Yuan and M. Medel, "Characterizing international travel behavior from geotagged photos: A case study of flickr," PLOS ONE, vol. 11, no. 5, 05 2016, pp. 1–18.

[7] C.-L. Kuo, T.-C. Chan, I.-C. Fan, and A. Zipf, "Efficient method for poi/roi discovery using flickr geotagged photos," ISPRS International Journal of Geo-Information, vol. 7, no. 3, 2018.

[8] Y. Yang, Z. Gong, and L. H. U, "Identifying points of interest by self-tuning clustering," in Proceedings of the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval, ser. SIGIR '11. ACM, 2011, pp. 883–892.

[9] V. Kulkarni, A. Moro, B. Chapuis, and B. Garbinato, "Extracting hotspots without a-priori by enabling signal processing over geospatial data," in Proceedings of the 25th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems, ser. SIGSPATIAL '17. ACM, 2017, pp. 79:1–79:4.

[10] X. Yang, Z. Zhao, and S. Lu, "Exploring spatial-temporal patterns of urban human mobility hotspots," Sustainability, vol. 8, no. 7, 2016.

[11] D. Laptev, A. Tikhonov, P. Serdyukov, and G. Gusev, "Parameter-free discovery and recommendation of areas-of-interest," in Proceedings of the 22Nd ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems, ser. SIGSPATIAL '14. ACM, 2014, pp. 113–122.

[12] M. Ester, H.-P. Kriegel, J. Sander, and X. Xu, "A density-based algorithm for discovering clusters in large spatial databases with noise," in Proceedings of the Second International Conference on Knowledge Discovery and Data Mining, ser. KDD '06, 1996, pp. 226–231.

[13] D. Comaniciu and P. Meer, "Mean shift: a robust approach toward feature space analysis," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 24, no. 5, 2002, pp. 603–619.

[14] D. Birant and A. Kut, "St-dbscan: An algorithm for clustering spatial-temporal data," Data & Knowledge Engineering, vol. 60, no. 1, 2007, pp. 208–221, intelligent Data Mining.

[15] "Openstreetmap," URL: https://www.openstreetmap.org/ [accessed: 2019-07-15].

[16] M. Shirai, M. Hirota, H. Ishikawa, and S. Yokoyama, "A method of area of interest and shooting spot detection using geo-tagged photographs," in Proceedings of The First ACM SIGSPATIAL International Workshop on Computational Models of Place, ser. COMP '13. ACM, 2013, pp. 34:34–34:41.

[17] M. Hirota, M. Shirai, H. Ishikawa, and S. Yokoyama, "Detecting relations of hotspots using geo-tagged photographs in social media sites," in Proceedings of Workshop on Managing and Mining Enriched Geo-Spatial Data, ser. GeoRich '14. ACM, 2014, pp. 7:1–7:6.

[18] D. J. Crandall, L. Backstrom, D. Huttenlocher, and J. Kleinberg, "Mapping the world's photos," in Proceedings of the 18th International Conference on World Wide Web, ser. WWW '09. ACM, 2009, pp. 761–770.

[19] S. Kisilevich, F. Mansmann, and D. Keim, "P-dbscan: A density based clustering algorithm for exploration and analysis of attractive areas using collections of geo-tagged photos," in Proceedings of the 1st International Conference and Exhibition on Computing for Geospatial Research & Application, ser. COM.Geo '10. ACM, 2010, pp. 38:1–38:4.

[20] M. Ankerst, M. M. Breunig, H.-P. Kriegel, and J. Sander, "Optics: Ordering points to identify the clustering structure," in Proceedings of the 1999 ACM SIGMOD International Conference on Management of Data, ser. SIGMOD '99. ACM, 1999, pp. 49–60.

[21] J. Sander, M. Ester, H.-P. Kriegel, and X. Xu, "Density-based clustering in spatial databases: The algorithm gdbscan and its applications," Data Mining and Knowledge Discovery, vol. 2, no. 2, 1998, pp. 169–194.

[22] J. Shi, N. Mamoulis, D. Wu, and D. W. Cheung, "Density-based place clustering in geo-social networks," in Proceedings of the 2014 ACM SIGMOD International Conference on Management of Data, ser. SIGMOD '14. ACM, 2014, pp. 99–110.

[23] Y. Chen, Z. Huang, T. Pei, and Y. Liu, "Hispatialcluster: A novel high-performance software tool for clustering massive spatial points," Transactions in GIS, vol. 22, no. 5, 2018, pp. 1275–1298.

[24] J. L. Bentley, "Multidimensional binary search trees used for associative searching," Communications of the ACM, vol. 18, no. 9, Sep. 1975, pp. 509–517.

[25] A. K. Mann and N. Kaur, "Survey paper on clustering techniques," International Journal of Science, Engineering and Technology Research, vol. 2, no. 4, 2013, pp. 803–806.

[26] M. Parikh and T. Varma, "Survey on different grid based clustering algorithms," International Journal of Advance Research in Computer Science and Management Studies, vol. 2, no. 2, 2014.

[27] R. Agrawal, J. Gehrke, D. Gunopulos, and P. Raghavan, "Automatic subspace clustering of high dimensional data for data mining applications," in Proceedings of the 1998 ACM SIGMOD International Conference on Management of Data, ser. SIGMOD '98. ACM, 1998, pp. 94–105.

[28] W. Wang, J. Yang, and R. R. Muntz, "Sting: A statistical information grid approach to spatial data mining," in Proceedings of the 23rd International Conference on Very Large Data Bases, ser. VLDB '97. Morgan Kaufmann Publishers Inc., 1997, pp. 186–195.

[29] C.-I. Chang, N. P. Lin, and N.-Y. Jan, "An axis-shifted grid-clustering algorithm," Tamkang Journal of Science and Engineering, vol. 12, no. 2, 2009, pp. 183–192.

[30] Y. A. Lacerda, R. G. F. Feitosa, G. A. R. M. Esmeraldo, C. d. S. Baptista, and L. B. Marinho, "Compass clustering: A new clustering method for detection of points of interest using personal collections of georeferenced and oriented photographs," in Proceedings of the 18th

Brazilian Symposium on Multimedia and the Web, ser. WebMedia '12. ACM, 2012, pp. 281–288.

[31] B. Thomee, "Localization of points of interest from georeferenced and oriented photographs," in Proceedings of the 2Nd ACM International Workshop on Geotagging and Its Applications in Multimedia, ser. GeoMM '13. ACM, 2013, pp. 19–24.

[32] M. Hirota, M. Endo, K. Daiju, and I. Hiroshi, "Discovering hotspots using photographic orientation and angle of view from social media site," International Journal of Informatics Society, vol. 10, no. 3, 2019, pp. 109–117.

[33] E. Spyrou, M. Korakakis, V. Charalampidis, A. Psallas, and P. Mylonas, "A geo-clustering approach for the detection of areas-of-interest and their underlying semantics," Algorithms, vol. 10, no. 1, 2017, p. 35.

[34] M. Omori, M. Hirota, H. Ishikawa, and S. Yokoyama, "Can geo-tags on flickr draw coastlines?" in Proceedings of the 22nd ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems, ser. SIGSPATIAL '14. ACM, 2014, pp. 425–428.

[35] Y. Hu, S. Gao, K. Janowicz, B. Yu, W. Li, and S. Prasad, "Extracting and understanding urban areas of interest using geotagged photos," Computers, Environment and Urban Systems, vol. 54, 2015, pp. 240–254.

[36] M. Chen, D. Arribas-Bel, and A. Singleton, "Understanding the dynamics of urban areas of interest through volunteered geographic information," Journal of Geographical Systems, vol. 21, no. 1, Mar 2019, pp. 89–109.

[37] Y. Zhu and S. Newsam, "Spatio-temporal sentiment hotspot detection using geotagged photos," in Proceedings of the 24th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems, ser. SIGSPACIAL '16. ACM, 2016, pp. 76:1–76:4.

[38] M. Shirai, M. Hirota, S. Yokoyama, N. Fukuta, and H. Ishikawa, "Discovering multiple hotspots using geo-tagged photographs," in Proceedings of the 20th International Conference on Advances in Geographic Information Systems, ser. SIGSPATIAL '12. ACM, 2012, pp. 490–493.

Mixed-Reality Communication System Providing Shoulder-to-shoulder Collaboration

Minghao Cai and Jiro Tanaka

Graduate School of Information, Production and Systems
Waseda University
Kitakyushu, Japan
Email: mhcai@toki.waseda.jp, jiro@aoni.waseda.jp

Abstract—In this paper, we propose a mixed-reality-based mobile communication system for two users placed in separate environments. The first is a remote user who physically travels to a shared environment with a mobile augmented reality setup, and the second is a local user who remains in another place while being immersed in a virtual reality view of the shared environment with the first user. The users are provided with a unique kind of collaboration, i.e., Shoulder-to-shoulder collaboration that simulates the users to walk shoulder-to-shoulder with viewing independence and bidirectional gesture communication. The major objective is to enhance co-located sensation. We introduce our prototype system as a proof of concept and perform evaluations of two user studies to verify system applicability and performance.

Keywords–Remote collaboration; Shoulder-to-shoulder; Viewing independence; Gesture; Co-located Sensation.

I. INTRODUCTION

In recent years, remote communication has been extensively used at the workplace and in everyday life to increase productivity and to improve the performance of instant communication. The advantage in allowing users from different locations to communicate and collaborate as a team helps the remote communication system become a cost-effective and popular way that can help users to get an instant solution for problems.

Although commercial remote conferencing technologies are cost-effective and more immersive than traditional phone calls that use only voice, most of these systems mainly provide a mere capture of both the user's face and limited transition in terms of body language or the reference of ambiance, which also act as a great source of information [1]. When indulging in a physical collaborative task or conversation with context related to the surroundings, existing technologies offer limited ways for users to achieve effective gestural communication, as they tend to focus on face-to-face interaction experiences. When users wish to describe the objects or directions in a scene or show operations, use of hand gestures would be more understandable than mere voice.

Another problem is in the form of the camera used for real-time video capture. When using telecommunication systems with smartphones or tablets, users tend to switch between the front and back cameras or they might place the device in a fixed position to attain a wider range of view. In most cases, the camera needs to be moved around for the remote person to perceive the entire scene. Such constraints make it difficult for users to get a common perception or to feel connected with each other.

Figure 1. Shoulder-to-shoulder communication for two users

In this paper, we propose a solution to these problems in the form of our prototype that provides a mobile *shoulder-to-shoulder communication system* for using mixed-reality (MR) collaboration and communication. This unique type of communication can enhance the user-to-user interactions and co-located sensation between users.

The prototype is designed for use by two users who are in different locations (as shown in Figure 1). For convenience, we refer to the user who goes to a remote environment, which would be shared, as the remote user, and the other one who stays in a local indoor workspace and remotely views the shared world as the local user, even though the roles may be reversed. We try to offer both the users with a shared feeling that they are going shoulder-to-shoulder together using gesture communication. Wearing a head-mounted display (HMD) with a virtual reality (VR) experience, the local user perceives the remote environment with viewing independence, while the remote user wears a see-through smart glass for an augmented reality (AR) experience.

To address the existing problems, as mentioned earlier, we create the following design requirements for the shoulder-to-shoulder communication prototype:

1   Offer the local user an independent view of the remote environment with control of his or her own viewpoint.

2   The local user should be able to easily see the remote partner's action and the direction of attention.

3   Offer an appropriate visual representation of the local user so that the remote AR user is aware of the attention and a improvement of the understanding and fidelity of the remote communication.

4   Provide mutual free-hand gesture communication

5   Offer visual assistance cue to enhance user interactions

The main contributions of this work are as follows:

- The design of shoulder-to-shoulder collaboration and a software system that supports MR collaboration between two users.

- The implementation of a prototype as a proof of concept (POC) that includes mobile setup for the remote VR user and a wearable setup for the local AR user.

- An evaluation consisting of two user studies to test the usability of the proposed prototype and user performance.

In Section II, we introduce related works. In Section III, we introduce our shoulder-to-shoulder collaboration and the corresponding system design. In Section IV, we introduce the implementation of our prototype. In Section V, we describe the evaluation that consists of two user studies in which we compare our should-to-shoulder communication design against two comparative conditions and then, test the system in a practical scenario. In Section VI, we discuss potential applications. In Section VII, we draw our conclusion to this work.

II. RELATED WORK

A. Remote Communication for Users Located in Different Places

Currently, it is not unusual to get instant contact with the use of commercial video conferencing systems (e.g., Skype and Cisco WebEx Conferencing). Most of these systems provide remote communication with a face capture feature from disparate locations, however, they do not allow users to reference a common physical ambient or share a co-presence feeling. Previous research has tried to address this limitation with different approaches [2] including projecting interface [3] and virtual reality interface [4].

Several pieces of research have made a lot of effort in working toward remote video communication techniques that aim at realizing a remote collaborative work experience among users in separate places [5, 6]. Some of these works tested depth sensors to extract and analyze body motions and interactions to support users to work in the same media space.

B. Remote Collaboration with Mixed Reality

Since the emergence of technology that supports remote communication [7, 8], researchers have started exploring remote collaboration with different degrees of user-to-user interactions. Reality is the user perception of the real environment. Introduced as a mix of both augmented reality and augmented
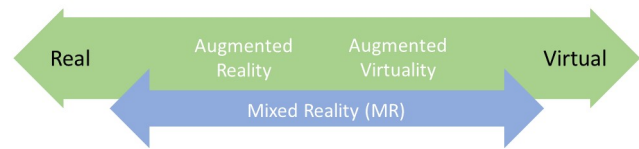
Figure 2. Virtuality continuum

virtuality (Figure 2 illustrates the reality continuum [9]), recently mixed-reality technique has been proven valuable for applications that involve a single user. It is believed that MR applications can provide users with a seamless combination of the virtual world and the real, physical world along with an enhancement of reality, which are the two major issues in traditional computer-supported collaborative work (CSCW). Researchers have started exploring the introduction of MR technology when constructing remote collaborations, where the VR representation of a local physical task environment is captured and shared with a remote-access collaborator who can view it in a separate remote place. Apart from verbal communication, the remote-access user is allowed to communicate with the local user with a certain degree of visual interactions.

Some previous works explored the use of pre-prepared 3D models to build virtual representations of the physical environment [10], or share 3D reconstructions of remote physical scenes on the user's desktop computer or mobile device [5]. Although these prior attempts with the use of static 3D reconstruction techniques can provide the spatial structure of the physical environment, they have limitations in terms of updating dynamical changes in the shared media space and moreover, in this case, the visual quality is typically inferior to a video image.

C. View Sharing in Remote Collaboration

A few researchers have tried using live video streams to share a view in an MR remote collaboration. Prior prototypes used handheld controls or touch screens to help a user create notes or draw annotations as visible cues in a 2D video stream [11–13]. Some system tried to introduce eye gaze or gesture [14] to improve communication efficiency. However, most of these works provide an egocentric viewpoint through a 2D video stream, in which the remote-access user's perspective was dependent on the motion of the camera capturing the surroundings.

Different approaches have been proposed to overcome the viewpoint limitation. Nuernberger et al. [15] demonstrated a system saving keyframes of the scene for later viewing. Fussell et al. [16] tried to place a camera fixed in the environment for remote collaboration. Some other attempt works explored utilizing remote presence and robotic techniques to offer remote-access users a certain control of the camera [17–20], but these approaches still have limitations with the field of view and delay in remote controlling the view.

Other researchers investigated using 360 panoramas [21] to help the remote-access user get a much larger field of view, or sharing panoramic images as an enhancement element of the 2D video streams in collaboration. In a demonstrated system, the researcher used a 360°camera to share the user's surroundings to a remote viewer who used mobile devices to

access an egocentric view [22]. They found the remote-access user had difficulties in communicating location and orientation information due to the lack of sharing gestures and other non-verbal communication cues.

D. Gesture Interaction

Hand gesture has been shown as an irreplaceable part for conversation, as it is treated as a cognitive visible awareness cue and provides rich contextual information that other body cues cannot reveal, which contributes significantly to a recipient's understanding [23, 24]. Over the past several years, some researchers have paid attention to support gestural interactions in a shared media space using different approaches. A study confirmed that over a third of the users' gestures in a collaborative task was performed to engage the other users and express ideas [25]. Kirk et al. [26] demonstrated the positive effect of using gestures and visual information in promoting the speed and accuracy in remote collaborative activities. Another work by Fussell et al. [27] demonstrated that users tend to rely more on visual actions than on speech in collaborative work.

E. Depth-based Gesture Recognition

Some researchers began to explore the idea of conveying gestures over a certain distance. A prior work [28] explored sharing live images of captured the arm action of one side's user on a remote shared tabletop screen for gesture collaboration. The gesture interaction in this work is still limited and the system only provides 2D images of hands or arms without any structural depth information. Several systems have captured users' hands in 3D and shared hand embodiments in a shared media space [29, 30]. However, these works require both local and remote users to remain within specific areas, which constrains the applications.

With the development of wearable devices and tracking sensors, some researchers have started exploring the use of a combination of depth cameras and head-mounted devices in experimental designs to realize remote collaboration in a reconstructed virtual-reality environment [30, 31]. These systems provide virtual hand gesture cues either captured with a depth camera [5] or represented by virtual hand models [31].

Previously, using depth-based gesture recognition, we built a remote sightseeing prototype that supported gestural communication to realize a gesture communication between two separated users [32, 33]. It was investigated by providing users with an approach to achieve a spatial navigation and direction guidance during mobile sightseeing. The positive evaluation results of this work encouraged us to support a mid-air gesture interaction for improvements to users' interactions in remote collaborations.

III. SHOULDER-TO-SHOULDER COLLABORATION

In this section, we introduce our proposed shoulder-to-shoulder collaboration and the system design that supports MR collaboration. This section consists of the following main aspects:

A   Overview of the prototype system
B   Shoulder-to-shoulder viewing independence
C   Shoulder-to-shoulder Gesture Communication
D   Tele-presence of the Local User's Head Motions
E   Virtual Pointing Assistance

Figure 3. Prototype overview

A. Prototype overview

Figure 3 illustrates an overview of the prototype. Shoulder-to-shoulder communication is an MR collaboration that provides shoulder-to-shoulder viewing independence and shoulder-to- shoulder gesture communication between two users. A remote user wears AR smart glasses and carries a 360 camera for capturing the remote environment. The 360°view is shared with a local user via the Internet. The local user utilizes an HMD as the display to observe the remote view and attains an immersive VR feeling. A depth camera is used to capture the local user's hand gestures for mutual gesture interactions.

B. Shoulder-to-shoulder Viewing Independence

To capture and share the real-time remote environment, we choose a 360°camera that provides a high-resolution video with a range of 360°in both horizontal and vertical directions. Unlike previous view sharing systems, where the camera was usually put on the remote user's head or cheek [34], our chosen 360°camera is mounted on the remote user's shoulder using a holder. The real-time 360°video is streamed back to the local site via the Internet and displayed in the HMD worn by the local user.

As the camera is fixed to the shoulder, its orientation is prevented from being influenced by the remote user's head movements. The local user has independent control over the viewing direction that can be manipulated by the head movements. As shown in Figure 4, the local user can simply turn the head to naturally change the viewpoints. Using this design, the local user immerses in the virtual remote world and perceives a sensation of personally standing next to the remote user and viewing the same scene.

C. Shoulder-to-shoulder Gesture Communication

In our proposed system, we provide the users with an approach to achieve a bidirectional gesture interaction during mobile communication. On one hand, a shoulder-looking capture of the hand gestures of the remote user is included in the local user's virtual viewing. On the other hand, a pair of virtual hands based on the depth-based recognition reappear during the local user's gestures in the remote user's field of view.

Figure 4. Independent control of the viewing direction for the local user
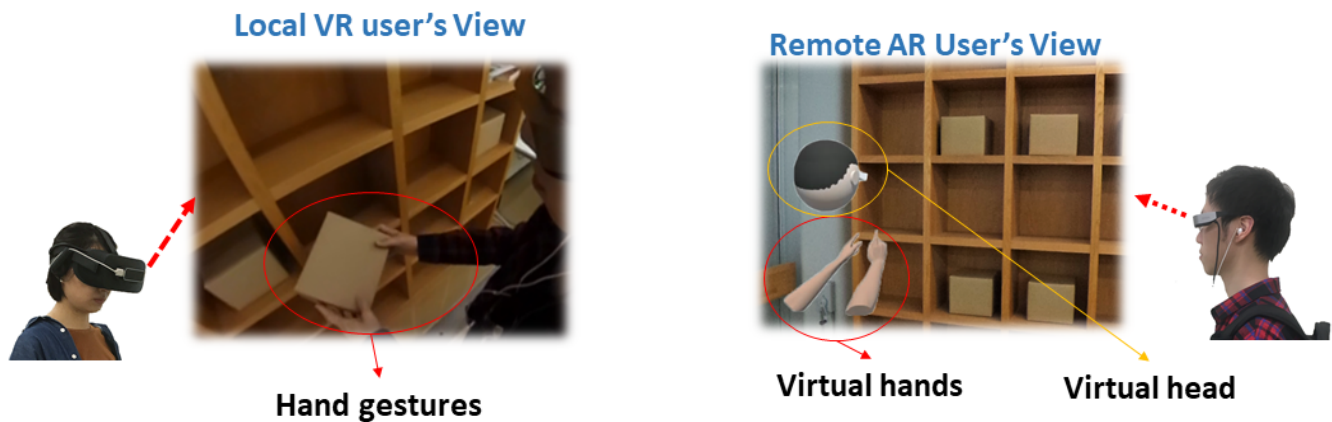
Figure 5. Local user's field of view: the remote user is making gestures

Figure 6. Remote user's field of view: the local user is making gestures. Red circle shows the virtual hands and yellow circle shows the virtual head representing the local user

1) Remote Gestures to Local User: As introduced in Section III-B, the local user has a 360°independent viewing of the remote world with a visual perspective obtained from the remote user's shoulder. This design allows the local user to see the remote user's hand gestures as well as the profile face. As shown in Figure 5, the local user simply looks leftward, and can directly see the remote partner performing hand gestures with an object (grabbing a box using the hands).

2) Local Gestures to Remote User: One of the important contributions of the proposed system is the reappearance of the local user's hand gestures in the remote world, as the local user is in a physically separate environment. We implement the hardware to extract the user's hand motion and the software to render it in the remote user's see-through smart glasses. Being considered as an accurate and convenient way, depth-based recognition has been used in current researches for hand motion extraction [29, 35]. A depth sensor is attached to the front side of the local user's HMD to extract a fine 3D structure data of both hands in real time. The local user can perform hand gestures without any wearable or attached sensors on the hands, which improve the freedom of hand motions and comfort. The system extracts the raw structure data with almost 200 frames per second with the help of the Leap Motion SDK [36]. We construct a pair of 3D hand models, which

include the palms and the different finger joints. This pair of 3D hand models is matched with the latest hand structural data. Thereafter, the current reconstructed hands are sent to the remote side via the Internet and rendered in the remote user's AR smart glasses as an event to update the previous hands. Therefore, once the local user makes hand gestures, the models change to match the same ones, almost simultaneously appearing in the remote user's field of view as well (Figure 6).

D. Tele-presence of the Local User's Head Motions

As we aim to enhance a co-located sensation by improving the interaction between users, we try to help the users by letting them easily know where their partner is exactly looking. It would improve the efficiency of communication when the user tries to join the same field of view to find out common interests or initiate a discussion. As we introduced in Section III-B, the local user can easily tell the remote user's viewing direction in the virtual scene. As the local user is in a physically separated environment, we construct a virtual head model to show his/her head motions in the remote user's view.

A motion tracking sensor is used to extract the head motion that is used to rotate the virtual head model. In Figure 6, the
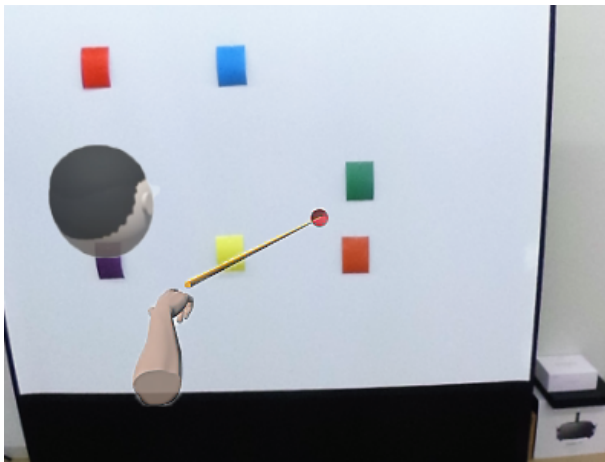
Figure 7. Remote user's view: Pointing cue for instructions

Figure 9. Local user's wearable setup: a Head-mounted Display with a depth sensor mounted on its front side

Figure 8. Zoomed in view of the pointing gesture

model present on the left side of the vision shows the remote user's precise facing direction.

E. Pointing Assistance

Previous research has shown that utilizing finger-pointing assistance can benefit cooperation and passing of instructions between users, especially when spatial information is involved in conversations [5].

In our shoulder-to-shoulder communication system, we allow the local user to use pointing assistance using fingers. The user performs a free hand pointing gesture that uses a virtual 3D arrow for showing the specific direction information in the remote user's view. This 3D arrow is treated as a spatial cue that assists a navigation or selection task during the communication (see Figure 7).

Our system uses a heuristic technique for gesture recognition. Using the depth sensor, our system can keep tracking the 3D structure of the user's hands including the different finger joints and can extract both the 3D position and orientation of the local user's fingers. The proposed system does not require calibration or precedent training. To activate the pointing technique, the user only needs to extend the thumb and index finger and keep the angle between them larger than the set threshold (see Figure 8).

IV. IMPLEMENTATION

A. Hardware

Our system's hardware involves two parts: the local user side and the remote user side.

1) Local User's Side: The equipment in the local user's side includes wearable devices (see Figure 9) and a desktop PC. The desktop PC (Intel Core i5, RX480 Graphics Card, 8GB RAM) on the local user's side is used to analyze data and as an engine for the core system. We use Unity as the engine to render and process the incoming data from both the remote and local sides, as well as to generate the graphical user interface (GUI) for both users. The headset that we chose as the local user's head-mounted display uses a pair of low persistence OLED screens, that provide a 110° field of view (FOV) [37]. A point tracking sensor is used to provide six total degrees of freedom in terms of rotational and positional tracking of the head movements. For hand motion tracking, the depth sensor used is light enough and introduces a gesture tracking system with sub-millimeter accuracy [38].

2) Remote User's Side: The integrated wearable device in the remote user's side consists of AR smart glasses, a 360° camera, and a notebook computer (see Figure 10). The AR glasses present a semitransparent display on top of the physical world, thus, allowing the user to view the physical world simultaneously. It is packed with a motion-tracking sensor for detecting the direction that the user is facing and a wireless module to exchange information with the local user's side via the Internet. It is also provided with an audio output with an earphone. The camera is connected to a notebook computer to generate a live stream so that the live video data can be sent to the desktop PC on the local user side using real-time messaging protocol (RTMP). The streaming uses an H.264 software encoder.

B. Software

We develop the software of our proposed system using Unity game engine [39] with Oculus Integration for Unity [40], Leap motion SDK [36], and MOVERIO AR SDK [41].
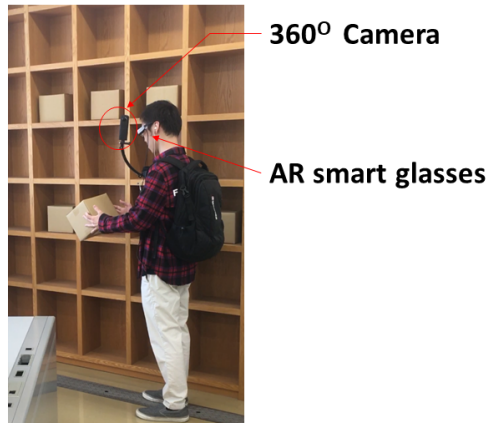
Figure 10. Remote user's setup: a AR smart glasses and a 360°mounted on the shoulder

Figure 11. Experimental workspace for Study 1

V. EVALUATION

In this section, we introduce our evaluation methodology that includes two studies to examine our Shoulder-to-shoulder prototype and test the design requirements, which were mentioned in the introduction section (Section I). In Study 1, we examine the effects of the viewing perspective of the proposed shoulder-to-shoulder viewing against two comparative conditions. This study assesses the design requirements for providing shoulder-to-shoulder viewing independence. In Study 2, we evaluate our proposed system in a more realistic collaboration. The purpose of this study is to investigate how the shoulder-to-shoulder communication affects the remote communication experience and co-presence sensation.

A. Evaluation Procedure

The two user studies are performed in the following order: Study 1→Study 2.

Before starting a study, the researcher explains all the equipment involved with the participants. The participants are then asked to try out the devices and fit the wearable equipment. At the beginning of each part, the researcher explains the purpose of the study and the participants' role in completing the tasks. The preparation time takes approximately 15 min for each participant. Further details about each part of the study are given in the following sections.

B. Study 1: Viewing Perspectives

In this study, we compare the different levels of viewing dependencies of the local user. We are interested in finding out how the difference in viewing perspective affects the remote-access user's spatial awareness level and social connection with the collaborators. Participants stayed indoors and did the test as the local VR user.

1) Workspace: In this study, we set up our experimental workspace in a room (see Figure 11). The workspace consists of a desk, a white partition, and a shelf which has multiple lattices. The local VR user and the remote AR user perform verbal communication over IP voice calls.

2) Participants: For our evaluation, we recruited 12 participants from our department. They were between the ages of 20 to 26 years. All participants possessed average computing skills and had some experience with AR or VR interfaces, which could reduce the novelty effect for the test results and will provide potential insight into our system from their experiences.

3) Study Design and Tasks: This study is a within-subject design, where we compare our shoulder-to-shoulder viewing with two other conditions (as shown in Figure 12(a)) of the local user's viewing perspective of a remote environment: (a)Dependent condition and (b)Stand-in condition.

- In Dependent condition, the participants, as the local user, use an egocentric viewpoint. The viewing perspective is dependent on the control of our researcher, the remote user. They see what the remote user see of the surroundings. In this condition, a capture of the surroundings is provided using a fixed forward camera of the smart glasses worn by the remote user, which always makes the viewer's viewpoint synchronously follow that of the recorder's. The participants browse the video in an HMD without viewpoint control.

- In Stand-in condition, participants, the local user, could see a 360°video of the workspace in a consistent orientation, viewing independently of the 360°camera's rotation. Under this condition, the capture of the surroundings is provided by a 360°camera mounted on the recorder's head

The whole study had two parts. In Part 1, the participants, as the local VR users, were asked to learn the remote surroundings under the direction of an actor, as the remote AR user, and figure out the object of interest that the remote AR user was randomly assigned to find. The object could be either a letter (on one of the boxes on the desk), a box (on a shelf), or color (on a partition). The remote user was not allowed to directly tell the participants what the object was, and they had to search the workspace together and find the object of interest as fast as they could. This task simulates a situation wherein it is difficult to verbally describe the spatial arrangement and the object of interest in the scene to the collaborator, for example, a workspace full of similar items.

During the test, the participants could ask the actor any binary questions that the actor could answer using "yes/true" or
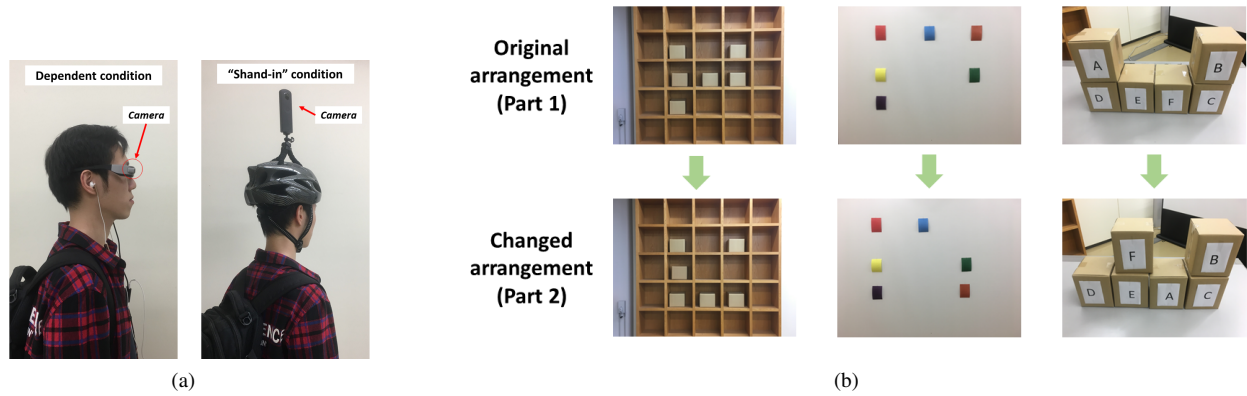
Figure 12. (a)Comparative conditions (b) Samples of changes of spatial arrangement task
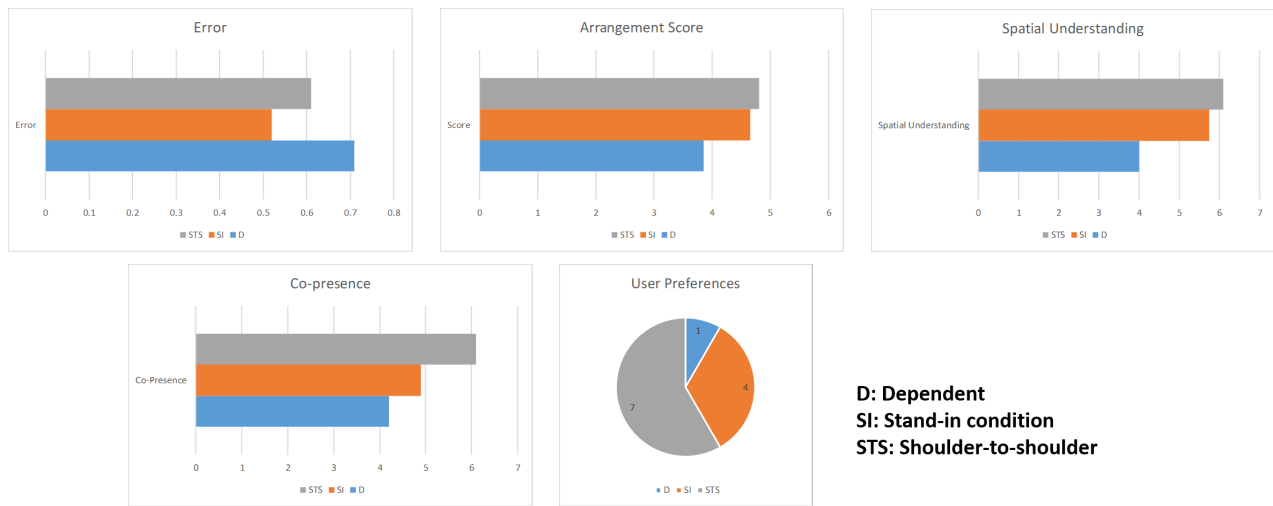
Figure 13. Results of Study 1.

"no/false". A wrong answer for each binary question asked was counted as an error. The participants were given two training trials for each condition.

In Part 2, to evaluate spatial understanding, the participants were given a spatial arrangement task. After a collaboration test in Part 1, our researcher randomly made 6 changes to the arrangement of the workspace by changing a set of the experimental objects' locations. To score test points, the participants had to move the objects back to match the original arrangement shown in Part 1 of the test.

Each participant was assigned four experimental trials for each condition and allowed a learning trial before the experiments. The order of conditions was counterbalanced between participants. The study took about 30 min for each participant.

4) Data Collection: We collected both objective and subjective data. The objective variables were the number of errors that occurred during Part 1 of the study and the score from solving the arrangement task in Part 2. The subjective measure was questionnaire consisting of Networked Mind Measure of Social Presence questionnaire [42] (on Co-Presence aspect), Spatial Understanding based on Spatial Presence Questionnaire

(MEC-SPQ) [43] with 6 item scale on Spatial Situation model (SSM), and user preferences. Social Presence and Spatial Understanding questionnaire were collected after each condition and user preference was collected at the end of the entire study. For all the conditions, our actor consistently performed the same way by looking toward the object of interest and utilizing hand gestures to assist verbal communication.

5) Hypotheses: We have the following hypotheses for this study:

H1 Higher degree of viewing independence (Shoulder-to-shoulder perspective or Stand-in perspective) increases the spatial understanding and lowers the subjective mental effort and task difficulty.

H2 Shoulder-to-shoulder perspective increases Social Presence score in terms of the Co-Presence (CoP).

H3 Participants prefer using perspectives that could provide a higher degree of viewing independence.

6) Results: In this study, we used the Friedman and Wilcoxon signed-rank test to analyze the significance of the experiment results across the three conditions. Figure 13 illustrates the results of Study 1.

a) Performance: From the result, we observe that the average number of errors, in all conditions, in Part 1 were below, indicating that there was almost no error in this part.

Figure 13 illustrates the mean test score of Part 2 for the three conditions. Pairwise comparisons yielded significant difference as the *Shoulder-to-shoulder condition* and the *Stand-in condition* performed much better than the *Dependent condition* (both p <0.01). However, no significant difference was found between the *Shoulder-to-shoulder condition* and the *Stand-in condition*.

b) Task difficulty: We found significant difference in pairwise comparisons between the *Shoulder-to-shoulder condition* and the *Dependent condition*, also between the *Stand-in condition* and the *Dependent condition*.

c) Spatial understanding: Pairwise comparisons yielded significant difference as the *Shoulder-to-shoulder condition* and the *Stand-in condition* got higher score than the *Dependent condition* (both p <0.01). However, no significant difference was found between the *Shoulder-to-shoulder condition* and the *Stand-in condition*.

d) Co-presence: In terms of Co-Presence aspect, we found significant differences as the *Shoulder-to-shoulder condition* performed much better than the other two conditions (all p<0.01). There was no significant difference between *Stand-in condition* and *Dependent condition*.

e) Preference: From the results, we found that, among three conditions of perspective, most of the participants preferred the *Shoulder-to-shoulder condition* (58%) followed by the *Stand-in condition* (33%).

f) Discussion: Our object and subject result strongly support our hypotheses H1, where the participants performed much better in spatial arrangement tasks and shown much better in the spatial understanding of the environment when they were provided with a higher degree of viewing independence. Our results also strongly support hypotheses H2 as there were significant differences in terms of Social Presence.

Most of the participants preferred having the shoulder-to-shoulder perspective and this supports our hypotheses H3, not only because it improved the users' confidence in spatial perception with view independence but also because it helped users in perceiving their partner's behaviors and hence, required less verbal communication.

C. Study 2 Collaborative Work

In this study, we evaluate the shoulder-to-shoulder communication under a more realistic collaboration scenario. The purpose of this study is to investigate how the shoulder-to-shoulder communication affects the remote communication experience and co-presence sensation.

1) Participants: For our evaluation, we recruited 12 participants from our department, which included six females. They were between the ages of 20 to 27 years with a mean age of 24 years. All of them possessed average computing skills and had some experience with AR or VR interfaces, which could reduce the novelty effect of the test results and also provide potential insight ino our system from their experiences. The participants were randomly grouped into six pairs. In each pair, one participant assumed the role of the local VR user, while the other participant assumed the role of the remote AR user.

TABLE 1. QUESTIONNAIRE

Q1. Did you observe interesting things independently?
Q2. Did you find it easy to tell your partner's viewing direction?
Q3. Did you feel gestural communication useful?
Q4. Did you feel the operation is easy enough to learn and use?
Q5. How much did you feel co-located with your partner during the test?
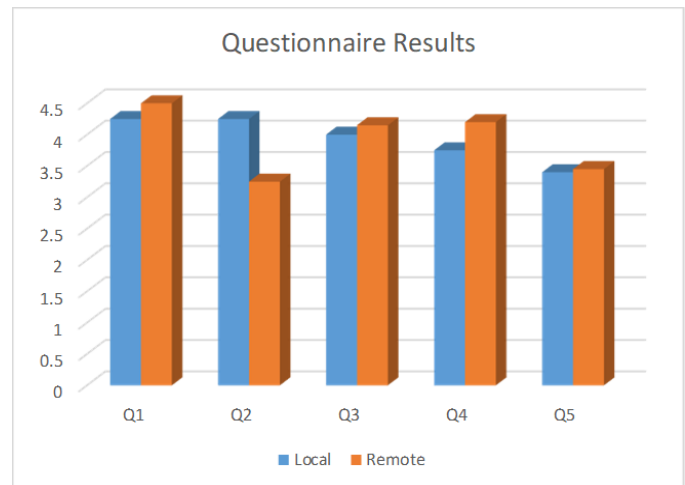
Figure 14. Questionnaire results

2) Study Design and Task: The experimental environment of the user study involved an indoor workspace for the local user and a departmental store, which was a larger space than the workspace used in Study 1, where the remote user stayed.

The task of this study was joint shopping. The goal for the participants was to work collaboratively and look for a product (such as a pencil box) that could interest both participants. In each pair, both participants were allowed a free voice communication supported by Internet IP phone call. The remote participant walked around and communicated with the local partner, and the local participant indulged in the shopping activity via remote communication. The subsystem used in the local user's part was connected to the cabled Internet, and the remote user's subsystem used a wireless connection (LTE). After the pilot test, we observed that the duration of completion was primarily influenced by personal preference. Therefore, we did not enforce any time limitation. This study was open-ended, and the only requirement was that the participants had to arrive at an agreement when selecting a product.

We collected subjective feedback from the post-task questionnaire. After each trial, the participants were asked to fill out a questionnaire that formed the basis of the subjective feedback. The participants graded each question using a 5-point Likert Scale (1 = very negative, 5 = very positive).

3) Results: Table 1 shows the questions that make up our questionnaire. We calculate the average score for each question in each group. Figure 14 shows the results. The results are divided into two groups–the local user's group and the remote user's group.

Question 1 – *Did you observe interesting things independently?* This question is used to test whether our system could
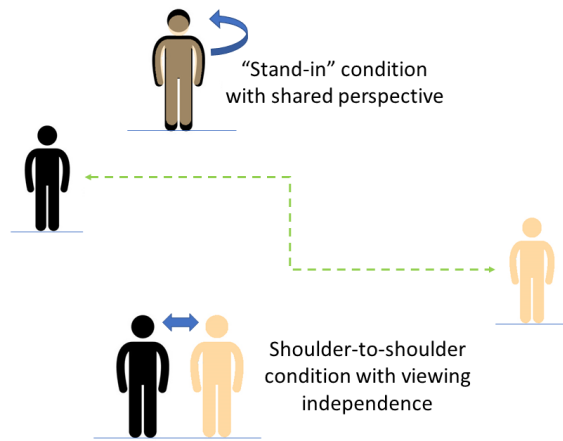
Figure 15. Comparison between the two types of remote communication

provide the users with viewing independence. According to the results, it was clear that both the users had independent control of viewpoint in remote view sharing.

Question 2 –*Did you find it easy to tell your partner's viewing direction?* This question indicates that the users could be aware of the partner's attention condition easily that made it possible for them to join in the same scene for further communication.

Question 3 – *Did you feel gestural communication useful?* This question is used to judge the practicability and effectiveness of the hand gesture communication through our system. It indicated that both the local user and the remote user found performing gestures to transmit their intentions usefully.

Question 4 – *Did you feel the operation is easy enough to learn and use?* This question is used to evaluate the ease of usability of our system. The result suggested that both the users generally found it effortless to achieve communication using our system.

Question 5 – *How much did you feel co-located with your partner during the test?* This question is aimed at investigating the overall performance and user experience. It demonstrates that during remote communication, both the users perceive a certain extent of co-located sensation.

4) Discussion:

a) Mutual Gesture: We also observed that the participants who played the role of local users graded slightly higher than their partners who played the role of the remote user. This difference, which indicates an incomplete equivalence of the gesture communication, benefits the local users more when compared to the remote users. After further communication with the participants during post-task interviews, we found that the difference was probably because the remote users could use hand gestures (such as touching, squeezing, or grasping) to interact with physical objects.

b) Shoulder-to-shoulder vs First-person Perspective: In traditional view sharing designs, which usually are found in previous computer-supported cooperative work (CSCW) [4], the local user mostly perceives the remote venue with the same field of view as that of the remote user. With such

setting of first-person perspective (FPP) of the content, the remote user acts more like a "stand-in" of the local user rather than as a communicating partner (see Figure 15). It might lead to misunderstandings, thereby limiting the natural communication between the users. In contrast, our shoulder-to-shoulder communication simulates a shoulder-to-shoulder togetherness, which provides both the users with more independence and allows them to focus more on mutual interaction. This could enhance a co-located sensation, which is also supported by our user study results. In this evaluation, all the participants successfully finished the tasks. In each pair, the local participant and the remote participant could reach an agreement and pick up a target object after discussion. Each user was aware of their partners during the task, which provided the users with a close connection. We confirmed that both the users could enjoy the communication experience and generally received a certain level of co-located feeling.

VI. Potential Applications

The potential applications of our shoulder-to-shoulder collaboration prototype are not limited to the scenario used in our case studies. Our system is also suitable for other remote collaborative works or remote assistance. For example, in case of an emergency assistance scenario, an expert (the local VR user) tries to assist a worker (the remote AR user) in manual operations to handle problems for the first time; or, people with inconveniences (local VR user) can continue to stay in a comfortable environment and at the same time get a virtual sightseeing with their friends (remote AR user) to enjoy accompanying moments and rich lifelogging.

VII. Conclusion

In this paper, we introduced our design and implementation of a shoulder-to-shoulder communication prototype that aimed at enhancing a co-located sensation between two users in separate environments. This prototype supports users with viewing independence and bidirectional gesture communication. We also described our evaluation to investigate the system's usability and user performance. In Study 1, we examined the effects of viewing independence for our shoulder-to-shoulder communication system against two other conditions. In Study 2, we evaluated our system in a more realistic collaboration. The results demonstrated that both sides of the users could effectively transmit instructions relating to the physical world and could achieve a smooth remote collaboration, and finally could receive a certain degree of co-located sensation.

References

[1] Minghao Cai and Jiro Tanaka. Remote shoulder-to-shoulder communication enhancing co-located sensation. In *The Twelfth International Conference on Advances in Computer-Human Interactions (ACHI 2019)*, pages 80–85, 2019.

[2] Keisuke Tajimi, Nobuchika Sakata, Keiji Uemura, and Shogo Nishida. Remote collaboration using real-world projection interface. pages 3008–3013, 2010.

[3] Pavel Gurevich, Joel Lanir, Benjamin Cohen, and Ran Stone. Teleadvisor: a versatile augmented reality tool for remote assistance. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 619–622. ACM, 2012.

[4] Shunichi Kasahara and Jun Rekimoto. JackIn head: immersive visual telepresence system with omnidirectional wearable camera for remote collaboration. *Proceedings of the 21st ACM Symposium on Virtual Reality Software and Technology*, 23(3):217–225, 2015.

[5] Rajinder S Sodhi, Brett R Jones, David Forsyth, Brian P Bailey, and Giuliano Maciocci. BeThere: 3D Mobile Collaboration with Spatial Input. *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems - CHI '13*, pages 179–188, 2013.

[6] Seth Hunter, Pattie Maes, Anthony Tang, Kori Inkpen, and Sue Hessey. WaaZam ! Supporting Creative Play at a Distance in Customized Video Environments. *Conference on Human Factors in Computing Systems*, page 146, 2014.

[7] Sara A. Bly, Steve R. Harrison, and Susan Irwin. Media spaces: bringing people together in a video, audio, and computing environment. *Communications of the ACM*, 36(1):28–46, 1993.

[8] "Put-that-there": Voice and Gesture at the Graphics Interface. *Proceedings of the 7th annual conference on Computer graphics and interactive techniques - SIGGRAPH '80*, pages 262–270, 1980.

[9] Paul Milgram and Fumio Kishino. A taxonomy of mixed reality visual displays. *IEICE TRANSACTIONS on Information and Systems*, 77(12):1321–1329, 1994.

[10] Ohan Oda, Carmine Elvezio, Mengu Sukan, Steven Feiner, and Barbara Tversky. Virtual replicas for remote assistance in virtual and augmented reality. In *Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology*, pages 405–415. ACM, 2015.

[11] Susan R Fussell, Leslie D Setlock, Jie Yang, Jiazhi Ou, Elizabeth Mauer, and Adam DI Kramer. Gestures over video streams to support remote collaboration on physical tasks. *Human-Computer Interaction*, 19(3):273–309, 2004.

[12] Steffen Gauglitz, Cha Lee, Matthew Turk, and Tobias Höllerer. Integrating the physical environment into mobile remote collaboration. In *Proceedings of the 14th international conference on Human-computer interaction with mobile devices and services*, pages 241–250. ACM, 2012.

[13] Steffen Gauglitz, Benjamin Nuernberger, Matthew Turk, and Tobias Höllerer. World-stabilized annotations and virtual scene navigation for remote collaboration. In *Proceedings of the 27th annual ACM symposium on User interface software and technology*, pages 449–459. ACM, 2014.

[14] Keita Higuch, Ryo Yonetani, and Yoichi Sato. Can Eye Help You?: Effects of Visualizing Eye Fixations on Remote Collaboration Scenarios for Physical Tasks. *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems - CHI '16*, pages 5180–5190, 2016.

[15] Benjamin Nuernberger, Kuo-Chin Lien, Tobias Höllerer, and Matthew Turk. Interpreting 2d gesture annotations in 3d augmented reality. In *2016 IEEE Symposium on 3D User Interfaces (3DUI)*, pages 149–158. IEEE, 2016.

[16] Susan R Fussell, Leslie D Setlock, and Robert E Kraut. Effects of head-mounted and scene-oriented video systems on remote collaboration on physical tasks. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 513–520. ACM, 2003.

[17] Joel Lanir, Ran Stone, Benjamin Cohen, and Pavel Gurevich. Ownership and control of point of view in remote assistance. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 2243–2252. ACM, 2013.

[18] Nobuchika Sakata, Takeshi Kurata, Takekazu Kato, Masakatsu Kourogi, and Hideaki Kuzuoka. Wacl: Supporting telecommunications using wearable active camera with laser pointer. In *ISWC*, volume 2003, page 7th. Citeseer, 2003.

[19] Abhishek Ranjan, Jeremy P Birnholtz, and Ravin Balakrishnan. Dynamic shared visual spaces: experimenting with automatic camera control in a remote repair task. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 1177–1186. ACM, 2007.

[20] Corey Pittman and Joseph J LaViola Jr. Exploring head tracked head mounted displays for first person robot teleoperation. In *Proceedings of the 19th international conference on Intelligent User Interfaces*, pages 323–328. ACM, 2014.

[21] VR QuickTime. An image-based approach to virtual environment navigation, shenchang eric chen, apple computer, inc. In *Siggraph, Computer Graphics Proceedings, Annual Conference Series*, pages 29–38, 1995.

[22] Anthony Tang, Omid Fakourfar, Carman Neustaedter, and Scott Bateman. Collaboration in 360 videochat: Challenges and opportunities. Technical report, University of Calgary, 2017.

[23] Charles Goodwin. Gestures as a resource for the organization of mutual orientation. *Semiotica*, 62(1-2):29–50, 1986.

[24] Susan Wagner Cook and Michael K Tanenhaus. Embodied communication: Speakers' gestures affect listeners' actions. *Cognition*, 113(1):98–104, 2009.

[25] John C Tang. Findings from observational studies of collaborative work. *International Journal of Man-machine studies*, 34(2):143–160, 1991.

[26] David S Kirk and Danaë Stanton Fraser. The effects of remote gesturing on distance instruction. In *Proceedings of th 2005 conference on Computer support for collaborative learning: learning 2005: the next 10 years!*, pages 301–310. International Society of the Learning Sciences, 2005.

[27] Darren Gergle, Robert E Kraut, and Susan R Fussell. Action as language in a shared visual space. In *Proceedings of the 2004 ACM conference on Computer supported cooperative work*, pages 487–496. ACM, 2004.

[28] Anthony Tang, Carman Neustaedter, and Saul Greenberg. Videoarms: embodiments for mixed presence groupware. *People and Computers XX—Engage*, pages 85–102, 2007.

[29] Judith Amores, Xavier Benavides, and Pattie Maes. Showme: A remote collaboration system that supports immersive gestural communication. *Proceedings of the 33rd Annual ACM Conference Extended Abstracts on Human Factors in Computing Systems*, pages 1343–1348, 2015.

[30] Franco Tecchia, Leila Alem, and Weidong Huang. 3d helping hands: a gesture based mr system for remote collaboration. *Proceedings of the 11th ACM SIGGRAPH International Conference on Virtual-Reality Continuum*

and its Applications in Industry, pages 323–328, 2012.

[31] Morgan Le Chénéchal, Thierry Duval, Valérie Gouranton, Jérôme Royan, and Bruno Arnaldi. Vishnu: virtual immersive support for helping users an interaction paradigm for collaborative remote guiding in mixed reality. In *2016 IEEE Third VR International Workshop on Collaborative Virtual Environments (3DCVE)*, pages 9–12. IEEE, 2016.

[32] Minghao Cai and Jiro Tanaka. Trip together: A remote pair sightseeing system supporting gestural communication. *Proceedings of the 5th International Conference on Human Agent Interaction*, pages 317–324, 2017.

[33] Minghao Cai, Soh Masuko, and Jiro Tanaka. Gesture-based mobile communication system providing side-by-side shopping feeling. *Proceedings of the 23rd International Conference on Intelligent User Interfaces Companion*, pages 2:1–2:2, 2018.

[34] Gun A Lee, Theophilus Teo, Seungwon Kim, and Mark Billinghurst. Mixed reality collaboration through sharing a live panorama. In *SIGGRAPH Asia 2017 Mobile Graphics & Interactive Applications*, page 14. ACM, 2017.

[35] Hani Karam and Jiro Tanaka. Finger click detection using a depth camera. *Procedia Manufacturing*, 3:5381–5388, 2015.

[36] LEAP MOTION. Leap Motion's SDK, 2019.

[37] Oculus. Oculus Rift, 2019.

[38] LEAP MOTION. LEAP MOTION, 2019.

[39] Unity 3D. Unity3D Game Engine, 2019.

[40] Oculus. Oculus Integration for Unity, 2019.

[41] EPSON. Moverio AR SDK, 2019.

[42] Frank Biocca Chad Harms. Internal Consistency and Reliability of the Networked Minds Measure of Social Presence. *Seventh Annual International Workshop: Presence 2004*, pages 246–251, 2004.

[43] Peter Vorderer, Werner Wirth, Feliz Ribeiro Gouveia, Frank Biocca, Timo Saari, Lutz Jäncke, Saskia Böcking, Holger Schramm, Andre Gysbers, Tilo Hartmann, et al. Mec spatial presence questionnaire. *Retrieved Sept*, 18, 2004.

Systematic Application of Domain-Driven Design for a Business-Driven Microservice Architecture

Benjamin Hippchen, Michael Schneider, Pascal Giessler
Sebastian Abeck

Cooperation & Management (C&M), Institute for Telematics
Karlsruhe Institute of Technology
Karlsruhe, Germany
{benjamin.hippchen, michael.schneider, pascal.giessler, abeck}@kit.edu

Abstract—Today's cloud providers open up new opportunities for software development. Unfortunately, however, not all existing applications are ready for operation in the cloud. One reason for this is usually the chosen architecture, which offers little flexibility like the monolithic architecture. In order to take up the possibilities in the cloud, a flexible architecture such as the microservice architecture is required. Developing with such an architecture is challenging and requires experienced team members. Especially the design of microservice-based applications is the challenge. Utilizing domain-driven design can be beneficial in breaking business functionality down into microservices. But also the use of domain-driven design requires a lot of experience, due to the lack of systematic. For this reason, we have created a systematic for the application of domain-driven design in the context of microservice development. The systematic accompanies the development team through the development process and supports them in the design and modelling of microservices. To cover the choreography of microservices, a new diagram called context choreography has been added. We also present UML profiles to support the modeling activities. Our systematic has shown over a longer period of time that it has a verifiable positive effect on microservice development.

Keywords–Microservice; Microservice Architecture; Domain-Driven Design; Context Map; Bounded Context.

I. INTRODUCTION

This article is an extended version of [1], which was published at SOFTENG 2019. The following aspects were added to the original work: (1) Consideration of event-driven communication within the context choreography; (2) Supporting the application of DevOps paradigms with the context map and context choreography. (3) Modeling of the context map; (4) UML profile for context map and context choreography. The digital transformation is in progress and organizations must participate; otherwise, they will be left behind. Existing business models have to be rethought and new ones created. Tightly coupled to the business model is the organization's application landscape. Thus, this landscape has also to be reimagined. Meanwhile, microservice architectures have established themselves as an important architectural style and can be considered enablers of the digital transformation [2]. Therefore, one major step towards a digital organization is the migration of legacy applications with monolithic software architecture into a microservice architecture. Afterward, the architecture must be maintained to provide long-lived software systems. However, neither the migration, design and development of a microservice architecture nor its maintenance are easy to achieve.

The structure of the new microservice-based application seems straightforward for the development team. Some microservices communicate with each other and deliver business-related functionalities over web application interfaces (web APIs). Another approach to communication is the event-based exchange of information. One also speaks of messaging [3]. The use of events is becoming more and more popular, as it provides a loose coupling in the context of microservices. However, at this point, the corresponding development team must ask itself decisive questions: How many microservices do we need? In which microservice do we put which functionality? Do we interact with third party applications? Domain-driven design (DDD) by Evans [4] provides important concepts which help answering these questions. As a software engineering approach, DDD focuses on the customer's domain and wants to reflect this structure into the intended application. The business and its business objects are the focus of each developing activity. Technical details, like the deployment environment or technology decisions, are omitted and do not appear in design artifacts. This is also a weak point of DDD, because from a certain point technical decisions are of great importance. An example of this is the operation of microservices, which is of a purely technical nature. DDD emphasizes the use of a domain model as a main development artifact: all relevant information about the domain, or business, is stored in it.

For microservice architectures, DDD helps structuring the application along business boundaries. Likely, these boundaries match the customer's domain boundaries. In his book *Domain-Driven Design: Tackling Complexity in the Heart of Software*, Evans introduces the "context map" diagram. The main purpose of the context map is to explore the customer's domain and state it as visual elements. The context map focuses for example on the macro structure of the domain, sub domains, departments instead of micro elements like business objects. A further essential DDD element and pattern is the "bounded context," which represents a container for domain information. This container is filled with the mentioned domain's micro structure, creating a domain model. The name bounded context is derived from its explicit boundary. Through this boundary, the container's content is only valid inside of the bounded context. From the strategic point of view, a bounded context is a candidate for a microservice. It is important to note, that the bounded context should be mostly independent of other parts of the domain. This supports the idea of microservices. Thus, the context map could display the organisation's microservice

architecture.

Like most DDD concepts, creating a context map is challenging and the tasks are not straightforward. The vague definitions and lack of process description impose problems. In addition, DDD does not provide a modeling language like Unified Modeling Language (UML). The following example illustrates several problems. A development team wants to establish a microservice architecture at Karlsruhe Institute of Technology (KIT) for the administration of students. Typically, for this purpose, universities introduce Student Information Systems (SIS) to support the business process execution for their employees. There are several problems with those SIS: (1) in the hands of software companies, (2) little to no understanding of the university's domain, (3) primarily monolithic architecture, and, (4) little to no insights for third parties. Since the development team has no affect on the SIS and its architecture, the goal is to enhance the SIS with social media aspects to support interaction between students. A microservice architecture is planned for the new functionality. In order to use DDD, the team must gather information about the domain and create a domain model and context map. The first uncertainty is the creation order of the artifacts. Both artifacts rely on information from each other. While creating the domain model, the development team has to know where to look for specific domain information, which is stated in the context map. When creating the context map, several bounded contexts are needed, which contain a domain model. In addition, the content of a context map is not precisely defined. The literature states that the context map contains bounded contexts and relationships but does not state how to elicit them or what they represent in the real world. This lack of real-world representation is especially a problem for development teams, who have to interact with an existing application. On the one hand, it is necessary to provide the third party application in the context map, since the context map can capture the information transferred between the third-party and the university. On the other hand, it is unclear how to represent the third-party application in the context map. A bounded context needs a domain model, and there is no domain model in this case. These are only two problems with the application of the context map, but they illustrate how import it is to enhance usability. In the following sections, we discuss these and other problems in more detail.

In this article, we provide the following contributions to enhance the application of the context map and support the design and maintenance of a microservice-based application:

- **Context Map Foundations**: One major problem of DDD is the lack of integration and placement in existing software development processes. It is unclear in which phases the context map must be created and in which phases it supports the development. Therefore, in Section III, we provide the first integration and placement of this map. In addition, we discuss the foundations of the context map and define the elements in this section.

- **Context Choreography**: While applying DDD for the development of microservice-based applications, we realized the existing artifacts did not capture all relevant information. Thus, in Section II and Section III, we introduce a new type of diagram, the "context choreography". This diagram's purpose is to display the choreography between multiple microservices for the application. To suit the ever more popular event-driven communication, the context choreography can represent the exchanged events between the microservices needed for the application.

- **Artifact Creating Order**: As mentioned, it is unclear in which order the DDD artifacts must be created. Therefore, in Section III, we also provide a detailed order with an emphasis on the context map. The application of the order is presented in our case study in Section V.

- **UML Profiles** - The look and feel of a context map depends on the modeler. DDD does not provide any modeling language for the style of a context map. Thus, each context map must be read differently, depending on the conceptional understanding of the modeler. We see a problem in the missing modeling systematic of a context map. In Section IV, we provide UML profiles for the context map and a the context choreography. Especially for the collaboration between multiple development teams, it is necessary to have a clear understanding and a limited amount of diagram elements.

II. PLACEMENT AND INTEGRATION OF THE CONTEXT MAP

One main problem of DDD is its lack of placement in the field of software development. Neither its models nor its patterns, including the context map, are placed in common software development processes. For our placement, we focus on the context of microservices. Since the context map has some weaknesses in development, a new diagram is introduced to close the gap.

A. Placement

As mentioned in Section I, the use of the context map is not straightforward. The development team must analyze the domain, create a domain model, and develop a context map. On the one hand, the context map has a great benefit for microservice architectures. On the other hand, applying the context map correctly is difficult.

Each DDD practice should be performed with the focus on an intended application [4]. This ensures the "perfect fit" of the gathered information, called "domain knowledge," for the application. Domain knowledge is captured in domain models. At this point, the pattern "bounded context" becomes important. An application consists of multiple bounded contexts, which all have their own domain models. With respect to the complexity of the domain knowledge, it makes sense to split the domain knowledge into multiple domain models. The validity of each domain model is limited by the bounded context. Furthermore, each bounded context has its own "ubiquitous language," which is based on the domain knowledge and acts as a contract for communication between project members and stakeholders. For the development of microservice-based applications especially, the multiple bounded contexts support the idea of a microservice architecture. Through connections between the bounded contexts, the domain knowledge is joined together in the application. The arising relationships are application-specific and differ from application to application. There are several types of relationships [5]. Modeling the

bounded contexts and their relationships is the purpose of a context map.

Considering a microservice architecture, the purpose of a context map is not only to elicit domain knowledge. Organizations that introduce microsevices need to manage their application landscape to maintain the microservice architecture. Without the knowledge about which microservices are available and who is in charge of them, the microservice architecture loses its advantages. Existing microservices are simply not used, even the domain knowledge they provide is required, due to the fact that other development teams could not find it, oversee it or forgot about it. The required domain knowledge is redeveloped in new microservices and the existing microservices become legacy. Sustaining the advantages of a microservice architecture is therefore important for organizations and the context map is one tool which helps to achieve this. In addition, the aspect that DDD focus its development artifacts on the customer's domain, supports the maintenance of the microservice architecture. Aligning the context map to the customer's domain leads to a natural-looking architecture [6]. Conway's Law [7] also supports the idea behind this. The organizational structure is adapted to the microservice architecture and vice versa. Looking at concepts like Martin Fowler's "HumaneRegistry" [8] or API management products like "apigee" [9], the idea and approach of the context map is required and furthermore it supports such concepts and products.

Using the context map as a tool for maintaining the microservice architecture is contradictory to one DDD aspect: focus always on an application. The mentioned maintenance does not require any kind of application-specific information. A microservice is mainly an application-independent software building block [10] and needs to be treated as such while maintaining the microservice architecture. Even if the development of a microservice is motivated through the development of an application. Thus, we see the context map as an application-independent diagram.
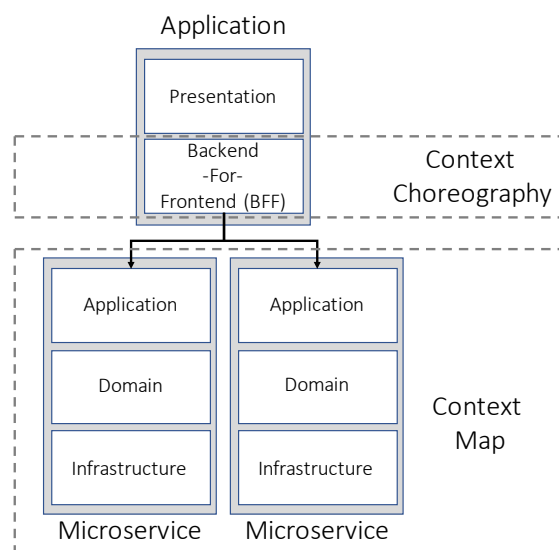

Figure 1. Placement of context mapping artifacts regarding the software building blocks from [10]

According to [10], for microservice-based applications, microservices are choreographed in applications through a backend-for-frontend (BFF) pattern. This is where application-specific information comes into play. Fig. 1 depicts the software architecture, including the application's BFF. To capture the choreography in the BFF, a new type of diagram is needed. The "context choreography" provides a view of the bounded context necessary for the application. Furthermore, the context choreography indicates which domain knowledge the bounded context transfers. We have chosen the term choreography because choreography is the real goal of the way Microservices communicate. A distinction is made between choreography and orchestration [11]. In orchestration, the interactions between the microservices must be triggered either by a central instance or by the microservices themselves. This results in an unwanted coupling between the building blocks. In contrast, the choreography relies on a loose coupling by reacting to events that take place in a message broker (such as RabbitMQ [12] or Kafka [13]). The microservices only need to know the events to which they have to react. It does not matter who created the event.

The context map can also be placed into software development activities. In [10], the first steps to place DDD into the software development activities from Brügge et al. [14] were taken. However, the context map itself was omitted. We built on these results for our placement of the context map. Domain-driven design introduces two types of "design activities" [4]. The first is the "strategic design," with tasks in modeling and structuring the domain's macroarchitecture (e.g., departments are used to define boundaries). This macroarchitecture is captured in the context map. Secondly, the "tactical design" further refines the macroarchitecture and enriches the bounded contexts with domain knowledge. This activity represents the microarchitecture of the domain and therefore of the microservice. Both activities rely on creating diagrams. Considering the software development activities from Brügge et al., Evans' designation as strategic and tactical "design" is misleading. Those focus more on the analysis than on the design phase. Many DDD practices and principles, such as "knowledge crunching," aim to analyze the domain. The development team explores the customer's domain and should simultaneously create the context map and domain model. Thus, the strategic and tactical designs are completed out, which is why the context map must be integrated at this point.

As mentioned, the content of the context map depends on its purpose. This is even the case for the relationships between the bounded contexts. Developing a monolithic application requires a different viewpoint on these relationships than a microservice-based application. A microservice architecture has many different microservices, which are managed by different development teams. By choreographing microservices in applications, development teams are automatically interdependent. This dependency is illustrated in the relationships in the context map. They could also be seen as communication paths between those development teams.

Our placement indicates that the context map has several possibilities to support the development of microservice architectures and microservice-based applications. We distinguish between a microservice architecture and the development of a microservice-based application. With regard to the microservice architecture style, the context map provides an overview

of all in the application landscape existing microservices and further the dependencies of the responsible development teams—which are also necessary information for maintaining the microservice architecture. Due to the application-independence of these information, the context map is an application-independent diagram. Additionally, we saw a lack of the context map while specifying microservice-based applications. Information transferred between microservices was missing a specification, which is necessary for choreographing the BFF of the application. Thus, we introduced the context choreography, which displays the application-specific dependencies between the microservices and their transferred domain knowledge. With this placement, we make a first step in advancing the use of the context map.

III. FOUNDATIONS AND ARTIFACT ORDER

In addition to the placement, we see a high need for clear definitions and guidance in creating the context choreography and context map. Therefore, this section provides definitions for terms regarding both artifacts. Afterward, we explain how the artifacts could be created.

A. Foundations

We found that, in addition to the development process, terminology around the context map is not clearly defined. This also leads to difficult application. Therefore, we want to provide some basics.

1) Bounded Context: The bounded context is the main element for the context map and is an explicit boundary for limiting the validity of domain knowledge [4]. Thus, within this context, there is a domain model and its ubiquitous language. A bounded context does not represent an application. This is based on the layered architecture of DDD, which consists of four layers: (1) presentation, (2) application, (3) domain, and (4) infrastructure. Domain-driven design and its artifacts focus only on the domain layer and omit the others. Therefore, without any application logic in a domain model, a bounded context cannot represent an application at all. This definition is fundamental, when creating a context map. An intended application is usually integrated into an existing application landscape.

When developing a microservice-based application, a bounded context initially only represents a candidate for a microservice [6]. Thus, a bounded context is either large enough that two or more microservices are necessary or small enough that they are included in one microservice. The best practice, however, should be the one-to-one relationship. This relationship eases the maintenance of the architecture through a clearer mapping between bounded contexts, microservices, and the responsible development teams. Reconsidering the size of the bounded context helps achieve this mapping. Therefore, we have collected several indicators, or more precisely possible influence factors, for the size of bounded contexts from our experiences in research and practice. This list should not be considered complete or verified with an empirical study but should rather be seen as an aid. A bounded context (1) has a high cohesion and low coupling, (2) can be managed by one development team, (3) has ideally a high autonomy to reduce the communication/coordination effort between development teams, (4) has a unique language that is not (necessarily) shared, and (5) represents a meaningful excerpt of the domain.

2) Context choreography: As mentioned (see Section II), the specification of a microservice-based application was lacking some information. Thus, we introduced the context choreography as a new diagram.

For microservice-based application development, it is important to state the other needed microservices—and thus also the bounded context. Furthermore, the exchanged data between those microservices are important information. As a microservice-based application is developed, existing microservices could still be used, while new ones are developed. In both cases, the context choreography is supportive. Regarding the application itself, the context choreography maps all microservices dependent on web interfaces and the consuming events, which are necessary to achieve the application's functionality. In further steps, the events to be published by the microservices could be integrated. A connection of the two microservices via an event does not mean that both communicate directly with each other. This is the case when shared entities are exchanged via web interfaces. According to the software architecture provided by [10], the application's BFF is specified. Independent from the application, the context choreography states the microservice web interfaces. Both the consumed and the provided interfaces of the microservice are provided. Thus, while developing the application, the first hints for designing the API can be derived. With regard to the subsequent maintenance of the microservice, development teams are able to identify the microservices that rely on them and vice versa.

3) Context Map: The DDD's original purpose for the context map differs from the one provided in this paper. In the context of microservice architecture, the context map is a useful diagram for maintaining the architecture and supporting application development.

One major advantage is the comprehensive overview of existing microservices. According to the best practice from Section III-A1, each bounded context in the context map represents a microservice. Further, in software architecture, social and organization aspects have to be considered [15]. Therefore, dependencies between microservices, and thus development teams, are stated. When development teams want to evolve their microservices, it is important to ask who depends on these microservices. At this point, the dependencies on other teams must be considered because any change could affect the stability of the other microservices. Thus, changes have to be communicated.

Also, for the development of a microservice-based application, the context map is advantageous. Regarding the context choreography, existing microservices are used to compose functionality for the intended application. Using existing microservices is only possible if they are traceable in the microservice architecture. This is where the context map comes into play. After developing a microservice, it is placed as a bounded context into the model. While the application is in development, the development team can use the context map as a tool to locate the needed microservices.

4) Domain Experts and other Target Groups: The interaction between domain experts and developers is one principle of DDD [4]. Each artifact is created for and with domain experts. Thus, the artifacts should be understandable without a software development background.

The context map according to DDD's definition is also relevant for domain experts [16]. However, according to our definition, we do not see any advantages for domain experts since the context map provides an overview of bounded contexts and communication paths between development teams. Furthermore, the context choreography is irrelevant. Only the subdomains, which represent the organization's structure, contain helpful information.

5) Supporting DevOps Paradigms with the Context Map and Context Choreography: The paradigm of Development & Operations (DevOps) is becoming more and more relevant for software development [17]. Especially in the context of the development of microservices, essential concepts have opened up. The paradigms in the context of microservices are relevant for coping with the challenges of the microservice architecture, such as the shorter time-to-market. The shorter development cycles can thus be absorbed by continuous integration and continuous delivery (CI/CD) [18]. A great synergy between microservice architecture and DevOps arises during the handling of the development team [19]. The development of microservices requires a rethink within the development team. The classic approach of large development teams is to split up into smaller self-sufficient development teams [6]. This idea is also supported by DevOps, which empowers a development team for the entire lifecycle of a microservice.

Important for the use of DevOps is the consideration of the paradigms during the architectural design [20]. For example, it is highly relevant that a microservice can be further developed independently of other microservices. To achieve this, the content of the microservices must be highly cohesive in order to be loosely coupled to other microservices. This is precisely where the designed system and the artifacts that are created come in handy. By using the context map, the domain can be separated into coherent microservices, which exist independently and can be further developed independently of the other microservices. Furthermore, the number of microservices and thus the number of necessary development teams can be read when designing the context map.

The context choreography can support the operations team of the microservices with the operation. From the artifact the necessary endpoints can be identified, which have to be released for the communication between the microservices. Such approval is particularly relevant at the infrastructure level.

B. Process for Establishing a Context Map

To develop a microservice-based application, it is necessary to establish the bounded contexts needed for the application. The developed application may reuse existing microservices, which should be integrated into the application landscape. To obtain an overview of the microservice landscape, the context map is useful. In this section, we focus on the establishment of the bounded contexts, the context choreography, and the context map. For developing an application, we build on a development process based on behavior-driven development (BDD) [21] and DDD [4] introduced in [10]. We omit the steps in [10] and focus on the creation of context choreography and a context map. Therefore, this section describes how the context map is established and enhanced within the development process.

1) Forming the Initial Domain Model: Forming the initial domain model occurs in the analysis and design phases. Before developing an application, the requirements are specified with BDD in the form of features. As Fig. 2 illustrates, a tactical diagram is derived from the features (e.g., the domain objects and their relationships). If a domain model already exists (e.g., from an existing microservice), this should be taken into account. The resulting diagram represents the initial domain model, which contains the application's business logic. Thus, the domain model provides the semantic foundation for all the specified features. The resulting diagram is comparable to a UML class diagram and displays the structural aspects of the domain objects. If the domain structure is still vague when the number of features is satisfied, more features are considered until the domain model appears to be meaningful. Afterward, as presented in Fig. 2, this initial domain model is examined and structured into several bounded contexts.
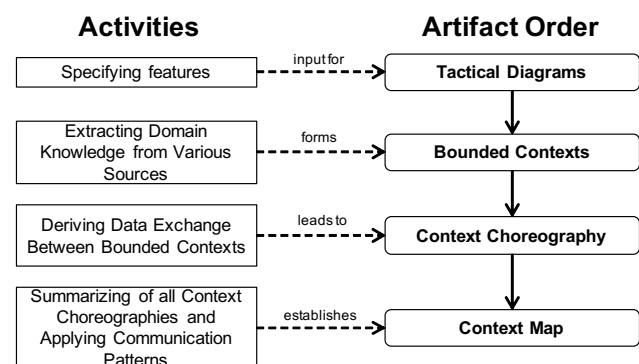
Figure 2. Creating order for artifacts their and impacting activities

2) Forming the Bounded Contexts: The model is strategical analyzed and separated based on the business and its functionality. This step depends on the domain knowledge and the structure of the business. Therefore, knowledge crunching from DDD [4] is applied to gather that knowledge. Often, a business's domain knowledge is scattered through the whole business. Therefore, analyzing the business is important to understand the business processes and the interaction of different departments. By default, each department knows its tasks the best. To extract the domain knowledge, various sources should be considered. These sources include domain experts who are part of a department, as well as documents and organizational aspects. This domain knowledge provides hints for structuring the domain and has to be considered while forming the bounded contexts. Considering the application analysis and the business analysis from [10] leads to the bounded contexts, as illustrated in Figure 2. If a context map has been established, then the context map is searched for the required domain knowledge of the application. If a bounded context representing the domain knowledge already exists, then this bounded context is taken into account. A new bounded context is established if the context map does not contain the required domain knowledge. For example, we integrated a profile context into an existing context map of the campus management domain.

3) Toward the Context Choreography: Forming the bounded contexts is only the first step towards a working

TABLE I. Overview of communication patterns and their impact

Comm. pattern	Description	Effort
Partnership	Cooperation between bounded contexts to avoid failure	Very high
Shared Kernel	Explicit shared functionality between different development teams	Medium to high
Customer/ Supplier	Supplier provides required functionality for the customer. The customer has influence on the supplier's design decisions	High to very high
Conformist	Similar to customer/supplier but with no influence on design decisions.	Low to medium
Separate Ways	No cooperation between development teams	Low
Anticorruption Layer	Additional layer that transforms one context into another	Low
Open Host Service	Uniform interface for accessing the bounded context	Low
Published Language	Information exchange is achieved using the ubiquitous language of the bounded context	Low

application. Each previously established bounded context is considered a microservice and requires or offers a unique interface for communication that can be based on REST or other paradigms like messaging. The microservices are choreographed with the BFF. Either the BFF directly accesses the web interfaces or creates events to which the microservices react. To allow choreography, the data exchange between the bounded contexts is considered next. The required data exchange is modeled in the context choreography. For each bounded context, a context choreography diagram is modeled. Domain objects that need to be shared or consumed from other bounded contexts are modeled as shared entities or events. The considered bounded context can either share the domain object itself or consume the domain object. Within an event there is information about the domain objects (see domain objects [5]).

4) Toward the Context Map: In microservice architectures, each established bounded context represents a microservice and is implemented by autonomous development teams. Thus, for relating bounded contexts, teams may need to communicate with each other. Therefore, the communication effort between the teams should be considered. The communication effort indicates how much communication between the teams is required. Clear communication paths are necessary, because a team needs to know which other team is responsible for relating bounded contexts. Therefore, dependencies and communication channels between teams are defined. Depending on the teams and the possible communication effort, a communication pattern is chosen based on [4] [5] (see Table I). The last three communication patterns listed in Table I are special patterns designed to reduce the communication between different teams, as well as the impact on interface changes. Other benefits and drawbacks of particular patterns exist but they are out of the scope for the current discussion. The context map illustrates the determined communication path between the bounded contexts. For example, when the communication between teams is not possible, such as when foreign services are adopted, DDD patterns need to be applied. For foreign services, the ACL pattern should be applied. In the last step, as depicted in Figure 2, the relationships (including the pattern) and the bounded contexts are added to the context map diagram.

Adding those bounded contexts and communication re-

lationships is an essential part of the context map. This concludes the first cycle of the analysis and design phases. The information about team dependencies helps in managing the development team and especially in restructuring in the sense of microservice architecture and DevOps.

5) Adjustments of the models: After the design phase, the implementation phase follows. In this phase, the models are implemented and tested. Afterward, specific parts of the application are developed. Following the iterative process, new features are implemented into the next cycle. These features need to be analyzed and may change the domain model. In addition, this may lead to the establishment of new bounded contexts. Thus, the models, including the tactical diagrams, the context choreography, and the context map, are refined according to the features and the knowledge crunching process in the previous steps.

IV. MODELING OF CONTEXT MAP AND CONTEXT CHOREOGRAPHY

In Chapter III we defined the foundations of a context map and also introduced the context choreography as a new type of diagram. Both types of diagrams have to be modeled. Considering the fact, that DDD does not provide any modeling guideline, the appearance—or the syntax—of a context map is broadly diversified. But also the semantics varies from model to model. As a consequence, comprehensibility and interoperability of development teams is reduced. Therefore, this chapter discusses the feasibility of creating a UML profile for the context map and context choreography. It will be integrated in the modeling application Enterprise Architect (EA), although any other software modeling tool like ArgoUML could be used alike.

A. UML Diagram Type and Elements

The purposes of the context map and context choreography are different, but for both it is necessary to express relationships between bounded contexts. On the one hand, the context map displays the way how teams interact with each other. On the other hand, the context choreography shows, which information is shared between bounded contexts. Both models are essentially component diagrams and share the concept of bounded contexts and relationships, but other diagram elements differ.

1) Context Map Elements: The context map provides an overview of all bounded contexts and their relationships. It is important to use an expressive and at the same time limited syntax. This should keep the complexity manageable and still cover all important business and domain information. In order to meet these requirements, the model should contain at least three types of visual elements. The essential diagram elements are components, stereotypes and associations which are shown in 5. Unlike components and associations, stereotypes can not exist on their own. They are an extension for adding specific information to an element. Additionally the use of more modeling elements, as packages for grouping components into subdomains and notes for general annotations, are advised. Components are used as containers for artifacts, which are in case of the context map a domain model. To distinguish their purpose they are tagged with stereotypes either as a "bounded context", "foreign bounded context", an "anti corruption layer (ACL)" or as a "shared kernel" [22]. The purpose of bounded contexts are already explained in Section III-A1.

a) Foreign Bonded Context: In comparison to a bounded context the foreign bounded context should be used for any bounded context whose domain knowledge is not fully accessible; just like a blackbox. Therefore, this the foreign bounded context might share an domain object which does not suit the needs of a bounded context. Neither responsibilities of development teams nor the content can be affected for a foreign bounded context. Without further treatment the domain knowledge of a bounded context must be adapted to this domain object which can affect its design in a bad way. Therefore the ACL supports a bounded context by converting domain objects from such a foreign bounded context into the required form. It is important, that no domain knowledge is packed into an ACL, which should be in the bounded context itself.

b) Anti Corruption Layer: The anti corruption layer (ACL) is used by the downstream as an isolating layer to communicate with the upstream. Hereby, the ACL converts the domain objects according to the domain models of those two bounded contexts. This means few or no changes to the domain model of the downstream have to be made. As mentioned earlier, the ACL does not provide any domain knowledge itself. It could be seen as an additional layer.

c) Shared Kernel: The shared kernel is the last essential element for the context map. It is used to share a subset of domain objects among multiple bounded contexts [5]. A reference from a bounded context to a shared kernel means that the domain knowledge of the shared kernel is included in the bounded context. As soon as two bounded contexts refer to a shared kernel, further developments of it has to be coordinated between the regarding development teams. This also leads to a shared responsibility.

d) Relationships: Associations are used to connect bounded contexts regarding their organizational structure and dependencies to other development teams. There are two types of associations. Firstly directed ones to represent an upstream/downstream relationship where the arrow points in the direction of the downstream. Secondly there are undirected ones to express that they are equal. To make the associations more expressive and thus more valuable they are tagged with additional stereotypes. Directed associations can be tagged as "customer/supplier" and "confirmist", while undirected ones can be tagged as "partnership" [5].

e) Omitted Associations: Literature introduces more patterns than these described above, but in our case, we did not see any advantages for them in the context of designing a microservice architecture. One pattern is called "separate way", which is used when two bounded contexts should stay independent of each other despite one could integrate the other to fulfill its functionality. It should be applied when there is an expensive integration of the regarding bounded contexts. But having microservices in mind, each microservice is independent by definition [6].

Another pattern is the "published language". Here, the upstream shares a well-documented language which describes how the communication is done and how the domain objects are structured. Because we assume a microservice architecture, every bounded context must have an API specification, like OpenAPI. Hence it is not necessary to model it.

Finally an "open-host service" defines a protocol to give access to multiple subsystems as a set of services. This can be done globally by an API management tool which makes it unnecessary to model it at every bounded context.

2) Modeling example for a Context Map: A simple context map containing all previously defined elements can be seen in Figure 3. It should give a good impression of what a more complex context map would look like. The bounded con-
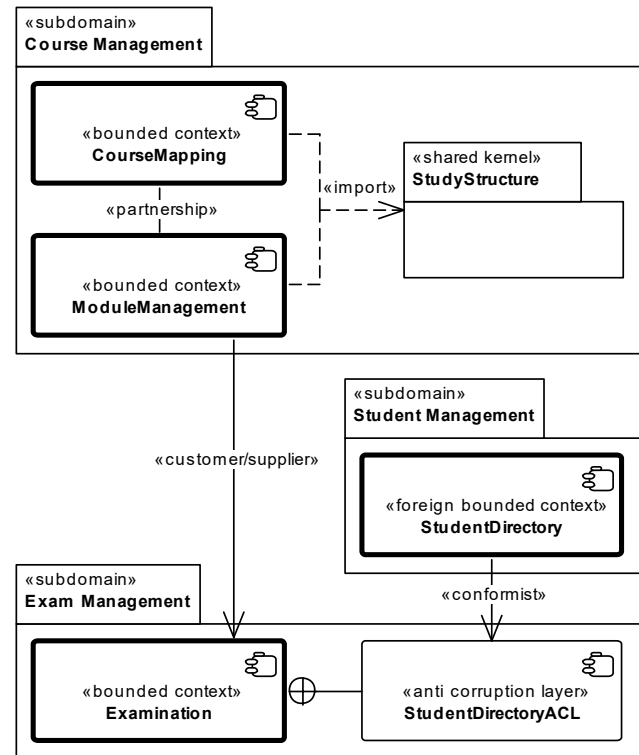
Figure 3. Example Context Map of the CampusManagement domain

texts, subdomains and relationships mainly serve exemplary purposes, which is why their meaningfulness is of secondary importance. Note that the anti-corruption layer must always be part of a bounded context, which is depicted with the UML nesting connector. Furthermore, the conformist relationship between *StudentDirectory* and *StudentDirectoryACL* can be understood as a direct relationship between *StudentDirectory* and *Examination* where *StudentDirectoryACL* is just an abstraction layer. The remaining elements and connections should be rather self-explanatory after the explanations above.

3) Context Choreography: Modeling the context choreography differs from the context map and is way easier to represent. Its purpose is to display communications between bounded contexts and further, the domain object, which is transferred by this communication. Therefore, we need bounded contexts, two types of associations and a shared entity as a new element. The bounded context works the same as for the context map.

a) Shared Entity: The shared entity or "*shared thing*" as Newman refers to it [6] is a domain object that is used for exchanging information between two bounded contexts. Thus, it is provided by one bounded context and consumed by another. It contains attributes and their types. Due to reusability and interchangeability, attribute types should be limited to the use of basic types.

b) Event: An event represents an event that occurs within a process flow. Usually, an event is generated when a domain object (especially the entities) has been manipulated. For example, if a new profile of a student was created, an event "StudentProfileCreated" could be thrown. The event contains all the information required for further processing of the process flow.

c) Association: The associations for the context choreography are simple. One connects the bounded context, which publishes the domain object, with the shared entity. It is tagged with "shares". The second one is to connect the shared entity with the consuming bounded context. It is tagged with "uses". The associations "produces" and "comsumes" are required for the events. The relationship between the event-generated bounded context and the event is labeled "produces". The consuming bounded context has a "consumes" labeled relationship.

B. UML Profiles for Context Map and Context Choreography

In case of both models, the syntax while modeling them has to be limited. Possible are UML profiles or metamodels, but each has their own advantages and disadvantages. Our approach to create a UML profile based on the UML component diagram. Further, we integrated it into a modeling tool named Enterprise Architect.

1) Metamodel vs. UML profile: Basically there are two options to lever UML for custom use cases. A lightweight extension describes the process of creating a UML profile with corresponding stereotypes, whereas an adaptation of the UML metamodel, by adding or changing existing elements, is called a heavyweight extension [23]. Usually, a lightweight extension is the better option because the majority of use cases can be covered, while keeping the complexity low. Additionally, the created UML profile is portable and can be used across most enterprise modeling applications like Enterprise Architect via the standardized XML Metadata Interchange (XMI) format. XMI, like UML, is defined by the Object Management Group (OMG). This open standard is supported by many software modeling tools and its main purpose is exchanging models, expressible in Meta-Object Facility, including their metadata. Finally, when extending UML via profiles, the modeling elements do not have to be relearned, because the base elements are the same and generally well understood.

The advantage of a heavyweight modification would be the even greater flexibility in mapping the modeling problem. It can be adapted to the most complex modeling requirements, which could not be mapped with a UML profile in a clean way. The downside is, that a custom modeling language, based on UML, has to be created, that every project member has to learn and understand, which usually is a time-consuming task to do. In addition, your modeling software has to support the change of the underlying metamodel, which is far less common than UML profile support.

In conclusion, the creation of a UML is preferable to an extension of the metamodel due to the low added value. Context maps, context relation views and domain models are rather intuitive and straightforward adaptations of existing diagrams, which are easily modeled by extending standard UML diagrams with stereotypes.

2) Enterprise Architect and Profiles: Enterprise Architect (EA) is a software modeling tool that is based on OMG's UML [24]. The tool allows the integration of user-defined extensions which includes the required UML profiles. In addition, EA already provides useful profiles for modeling languages, such as Business Process Modeling Notation (BPMN). Custom UML profiles can be created through EAs so-called Model Driven Generation (MDG) technologies. The UML profiles provided in this article are extended versions of EA's standard UML profiles. To create a UML profile with optimal user experience, additional diagram and toolbox profiles are necessary, which are specific to the EA. These custom diagrams extend standard UML diagram metaclasses and define the appearance of the diagram elements. A toolbox shows these defined elements to the user. Figure 4 displays an overview of the toolbox for the context map. The user sees only the modeling elements defined by the custom UML profile, which simplifies the modeling process and reduces modeling inconsistencies. We use EA to model the context map and the context choreography.
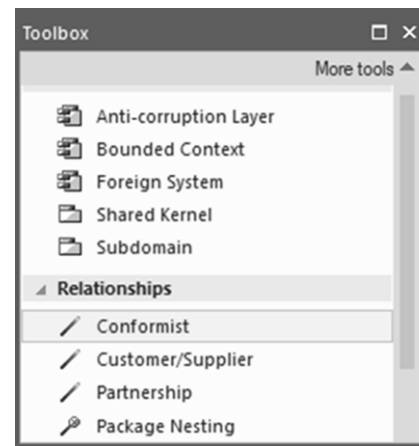
Figure 4. Custom toolbox for the context map

3) UML Profile for the Context Map: The core elements of our Context Map can be seen in Figures 5 and 6 and are the same that were defined earlier. Note that the metaclasses are only references to the elements in UML and are therefore not created by us, but only used by the stereotypes as a basis. The *direction* attribute of the association metaclasses specifies the default direction of a newly created association, whereas the *compositionKind* attribute specifies whether the association is an aggregation or a composition. Additionally, the standard UML package import connector was added to the toolbox for denoting shared kernel usage and is therefore not shown in Figure 6. It is a directional dependency relationship with the stereotype «import», starting from the using bounded context to the shared kernel.

4) UML Profile for the Context Choreography: Figure 7 shows the profile for the context choreography diagram. The metaclasses and the *direction* attribute behave in the same way as in the context map. The *isActive* attribute on the class specifies whether the class can operate as an independent thread of behavior. Besides the depicted elements in the profile, the associated toolbox does not contain any further elements, since these should be created in the context map and imported afterward.
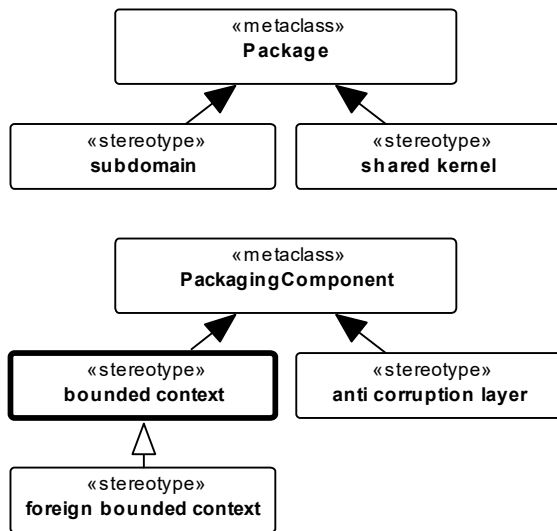
Figure 5. Core elements of the context map profile
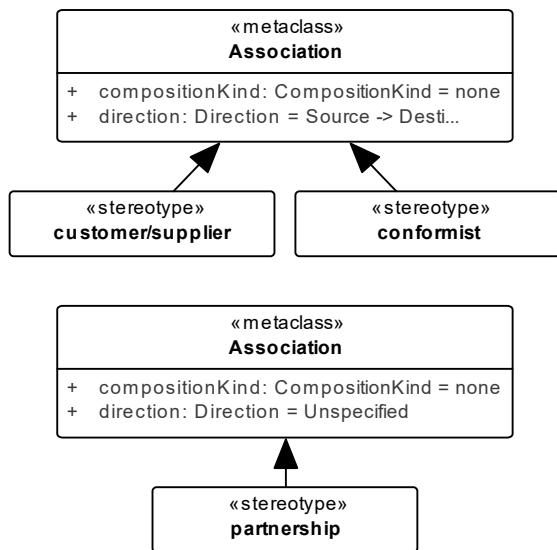
Figure 6. Excerpt of the available context map profile associations

Figure 7. Context choreography profile

For this purpose, we wanted to introduce a profile service in the campus management system. This profile service should allow students to create custom profiles to display information about their studies, like currently visited lectures and future exams. The purpose of this service is to assist students with their studies and their search for study partners. For example, students can find study partners with the help of the profile service when other students share the lectures they attend.

B. Requirement Elicitation

In our process, we began by eliciting the requirements with BDD and formulating them as features. Fig. 8 presents one of the main features that enables students to edit their profile.

1. **Feature**: Providing student profiles
2. As a student
3. I can provide relevant information about myself
4. So that others can see my interests and study information

5. **Scenario**: Publish profile
6. **Given** I was never logged in to the ProfileService
7. **When** I log in to the ProfileService for the first time
8. **Then** my study account is linked to the ProfileService
9. **And** I choose which profile information I want to publish

Figure 8. Example of a BDD feature for publishing an user profile

V. CASE STUDY: CAMPUS MANAGEMENT

In our case study, we illustrate our approach of establishing a new bounded context and integrating it into an existing context map. The case study orientates itself on the process as described in Section III. Over three semesters, we expanded the campus management system of KIT with microservice-based applications. The experiment carried out can be understood as type I validation [25]. With a type 1 validation, the testing of the design process can be seen in a realistic project context. It is legitimate to conduct the experiment with students as test persons [26]. The case study represents our recent project in this field and adds a social media component to the campus management system.

A. Project Scope

Our vision is to simply and efficiently support the exchange of information and facilitate cooperation between students.

C. Initial Domain Model

Analyzing the features leads to the initial domain model by deriving domain objects and their relationships. In our previously defined feature (see Section V-B), we identified, the terms "Profile," "Examination," and "Student" and added them

to the initial domain model. By repeating this procedure with all features, the domain model is enriched with the business logic. The result of the initial tactical diagram is presented in Fig. 9.

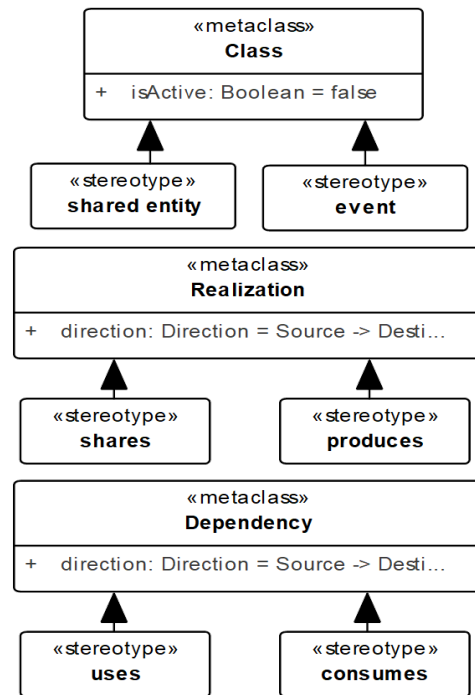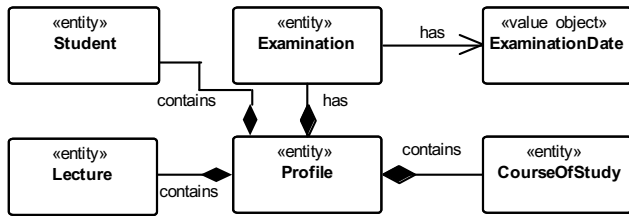Figure 9. Initial domain model derived from BDD features and other sources

D. Bounded Contexts and Context Choreography

While we analyzed the domain, we also considered the existing context map of the campus management domain. We noticed that the bounded contexts "StudentManagement," "ExaminationManagement," "ModuleManagement," and "CourseMapping" already offered the required functionality. Only "ProfileManagement" had to be established as a new bounded context. Therefore, we considered the data exchange between the bounded contexts and created the context choreography on that basis. The result is illustrated in Fig. 10. The existing bounded contexts provide the required data as shared entities. Furthermore, the relevant events for the process flow were considered. Unfortunately, not every third-party application is designed for communication with events. Therefore we still had to use the web interfaces. The new bounded context
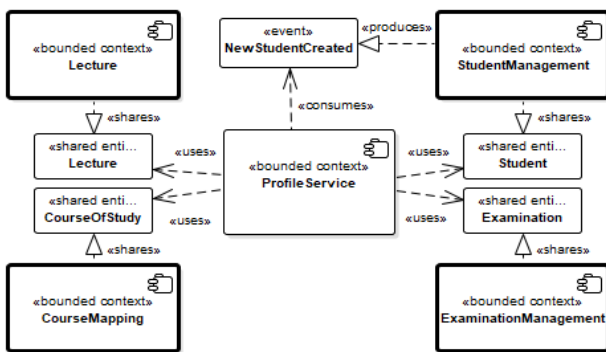

Figure 10. New bounded context "ProfileManagement" in a context choreography

"Profiles" adapts the shared entities and delivers the data required from the profile service. It also consumes the event "NewStudentCreated" of the "StudentManagement" Bounded Context. As soon as this event is consumed, a new profile based on the student can be created. Last, the microservices are choreographed in the BFF of the intended application, to achieve the required application logic.

E. Integrating in Context Map

After we had established the bounded contexts and the context choreography, we needed to add the profile management context to the context map. Therefore, we determined the

dependencies and communication channels between bounded contexts based on the context choreography. We found our development team did not influence any other bounded context. Thus, we applied the conformist as communication pattern. As a result, the context map depicted in Figure 11 was enhanced with the "Profiles" context. Afterward, we started the first implementation cycle.
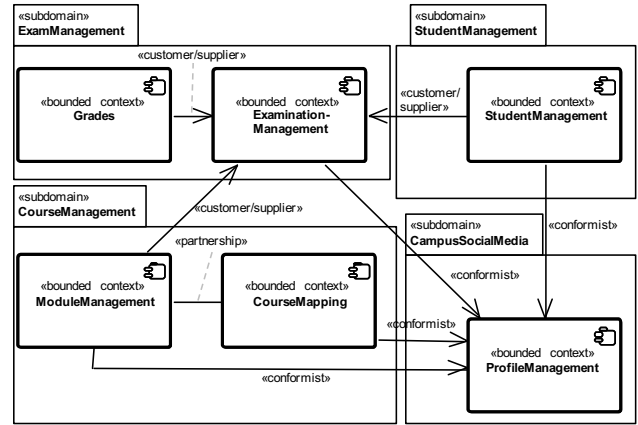
Figure 11. "ProfileManagement" context integrated into context map

F. Context Map as Template for a Deployment Map

The resulting context map provides an overview of the microservices that need to be deployed. Assuming that each bounded context represents one microservice, we can enhance the context map with technical information that is needed for the deployment in a secure manner. For instance, we can define which ports listen for incoming or are allowed for outgoing requests. By following such an approach, each microservice is initially considered in isolation. We enforce this by defining default policies on the execution environment that need to be taken into account during deployment. The enhancement with technical information is transferred into a new diagram called a deployment map. For the modeling, we use a UML deployment diagram. In addition to a general overview of the deployment, it can also be used for an upfront security audit. We are planning to present the derivation rules in an upcoming publication.

For testing purposes, we have used a hosted Kubernetes [27] cluster on a cloud provider. Kubernetes is an open source system for provisioning and management of container-based applications and aroused from the collected experience behind Omega and Borg. First of all, we have defined policies to deny all ingress and egress traffic to all running Pods by default. A Pod groups one or more containers and can be seen as the central brick of Kubernetes when deploying applications. Each bounded context will be represented by a microservice running in a container (Docker or rkt). Depending on their relation to each other (see TABLE I), we put them in corresponding Pods. Next, we have used the ports for incoming and outgoing traffic to derive the network policies. Finally, we have defined the services that wrap the Pods and offer a central access point for interaction. The application shows us that the underlying context map can be used as a basic scaffolding for deriving the deployment map but need to be

enhanced with technical information as well as information from the development teams that develop the microservices. For instance, a persistent storage is missing in a context map due to its domain orientation but is needed for the deployment map.

G. Organizing the artifacts in the Enterprise Architect

An essential part in software development is to retrieve design artifacts. To achieve this, a centralized repository is helpful. The EA innately provides such a centralized repository, so we were able to make use of it and easily retrieve our design artifacts. We placed each design artifact into a folder structure accordingly to our experiences from other projects [28]. The first artifact in the folder, as shown by figure 12 is

Figure 12. Repository structure of the campus management domain

the context map which has the same name as the repository. Folders of the different subdomains follow this artifact. Each subdomain folder can be expanded and contains the bounded contexts that belong to this subdomain. A bounded context itself contains the context choreography, the relation view and the domain objects which belong to the relation view.
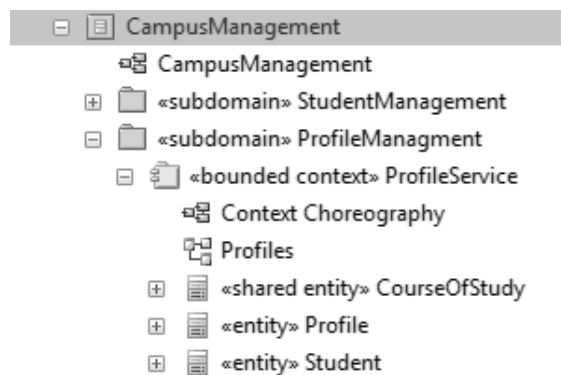
VI. RELATED WORK

During our research, we searched for works comparable to the context map and its purpose. We encountered several inspiring works regarding different aspects of the context map. Especially, the focus on the microservice architecture is an important part of this paper.

A. Microservice Architecture

A microservice should concentrate on the fulfilment of one task and should be small, so a team of five to seven developers can be responsible for the microservice's implementation [6]. A microservice itself is not an application, but rather a software building block [10]. In microservice architectures, applications are realized through choreography of these building blocks. A central aspect of microservice architecture is the autonomy of the single microservices [29]. Each microservice is developed and released independently to achieve continuous integration.

B. Approaches for Designing a Microservice Architecture

The objective of microservice architectures is to subdivide large components into smaller ones to reduce complexity and create more clarity in the single elements of the system [29]. In this article, we described our approach of designing

microservice architectures with a context map from DDD. However, there are further strategies to identify microservices, which we considered in this article.

One possible approach is event storming, as introduced by Alberto Brandolini in the context of DDD [30]. Event storming is a workshop-based group technique to quickly determine the domain of a software program. The group starts with the workshop by "storming out" all domain events. A domain event covers every topic of interest to a domain expert. Afterward, the group adds the commands that cause these events. Then, the group detects aggregates, which accept commands and accomplish events, and begins to cluster them together into bounded contexts. Finally, the relationships between bounded contexts are considered to create a context map. Like our approach, this strategy is based on DDD and results in a context map displaying the bounded contexts. Instead of a workshop for exploring the domain and defining domain events, we develop our bounded contexts through an iterative analysis and design phase. Furthermore, we enhanced the context map with maintenance aspect for microservice discovery and dependencies between development teams. The purpose of the resulting context map from [30] is comparable to the context choreography. Both focusing on the interactions between bounded context and identify the transferred domain knowledge.

Another method for approaching a microservice architecture is described in [31]. First, required system operations and state variables are identified based on use case specifications of software requirements. System operations define public operations, which comprise the system's API, and state variables contain information that system operations read or write. The relationships between these systems operations and state variables are detected and then visualized as a graph. The visualization enables developers to build clusters of dense relationships, which are weakly connected to other clusters. Each cluster is considered a candidate for a microservice. This bears a resemblance to our approach because we also begin by focusing on the software requirements and take visualization for better understanding the domain.

A further widely used illustration of partitioning monolithic applications is a scaling cube, which uses a three-dimensional approach as described in [32]. Here, Y-axis scaling is important because it splits a monolithic application into a set of services. Each service implements a set of related functionalities. There are different ways to decompose the application which differ from our domain-driven approach. One approach is to use verb-based decomposition and define services that implement a single use case. The other possibility is to partition the application by nouns and establish services liable for all operations related to a specific entity. An application might use a combination of verb-based and noun-based decomposition. X-axis and Z-axis regards the operation of the microservices. The X-axis describes the horizontal scaling which means cloning and load balancing the same microservice into multiple instances. Meanwhile, the Z-axis denotes the degree of data separation. Both axis are important for microservice architectures and currently omitted in our context map approach.

C. Software Development Approaches

The development process we apply is based on BDD and DDD. As a method of agile software development, BDD

should specify a software system by referencing its behavior. The basic artifact of BDD is the feature, which describes a functionality of the application. The use of natural language and predefined keywords allows the developer to define features directly with the customer [33]. During our analysis phase, we used BDD for requirement elicitation.

In our design phase, we applied DDD based on the features we defined with BDD. DDD's main focus is the domain and the domain's functionality, rather than technical aspects [16]. The central design artifact is the domain model, which represents the target domain. In his book *Domain Driven Design - Tackling Complexity in the Heart of Software*, Eric Evans describes patterns, principles, and activities that support the process of software development [4]. Although DDD is not tied to a specific software development process, it is orientated toward agile software development. In particular, DDD requires iterative software development and a close collaboration between developers and domain experts.

D. Application of the Context Map

The goal of a context map, which Evans describes as a main activity of DDD, is to structure the target domain [4]. For this purpose, the domain is classified into subdomains, and in those subdomains, the boarders and interfaces of possible bounded contexts are defined. A bounded context is a candidate for a microservice, and one team is responsible for its development and operations [6]. Moreover, the relationships between bounded contexts are defined in a context map. Both the technical relationships and the organizational dependencies between different teams are considered.

A further aspect of the context map involves the maintenance of the miscroservice architecture. Without managing the application (or service) landscape, existing microservices are not used, even if they provide needed domain knowledge. The usage of a context map helps concepts like humane registry or API management products which tries to achieve maintenance goals. Martin Fowler introduced humane registry as a place, where both developers and users can record information about a particular service in a wiki [8]. In addition, some information can be collected automatically, e.g., by evaluating data from source code control and issue tracking systems. API management products like "apigee" [9] reach maintenance by pre-defining API guidelines such as key validation, quota management, transformation, authorization, and access control.

E. Unified Modeling Language Profiles in the Context of Domain-Driven Design

How to model the artifacts of the DDD is a known problem. Depending on the experiences and the understanding of the modeler, the appearances of the domain model differs. This poses a problem for the inter-team communication, due to the fact, that each team possibly interprets modeled concepts different.

Based on the inter-communication problem, [34] provides a survey on DDD modeling elements together with an initial UML profile for the domain model. Their survey is based on the examples shown in Evans DDD book [4]. It shows that the most examples are based on informal UML diagrams, which prevents automatically validation, transformation and code generation. The UML profile is applied in the context

of microservice architectures with the goal of deriving code for microservices.

We see [34] as an important work for the application of DDD. Their results inspired us to also create an UML profile for the context map and context choreography.

VII. LIMITATIONS AND CONCLUSION

The concepts we provide in this article have some limitations. These are addressed in the next section. Afterward, we provide a short conclusion discussing our results.

A. Limitations

Domain-driven design has no special application or architectural style in mind. The concepts should be applied to DDD's layered architecture but could be applied to different architectural styles. For a better fit while developing microservice-based applications, we always had the microservice architecture in mind. Therefore, our provided concepts are only valid when developing a microservice-based application.

The concepts provided by this article are built from our experiences which we gathered in various projects. Most of our projects were in the academic branch, but we also worked with industrial partners. For the context map, we developed and proved our concepts over a longer time. Especially the provided UML profiles reflects our needs from the mentioned projects. It is likely, that the shown profiles are incomplete and need to be adjusted for further projects. The case study describes our last project. Project members and partners gave us useful feedback about the concepts when they applied them. In addition, the feedback also included points we had not yet addressed, like a modeling language for context mapping. Nevertheless, evidence of our concepts in large microservice architectures, such as 50 or more microservices, still lacks. Our goal is to obtain prove for large microservice architectures in such projects.

Another limitation of our concept is that we only applied them in "clean" microservice architectures. However, in the real world, there are also legacy applications in the application landscapes of organizations. Typically, a legacy application is not a microservice-based one; often, it is a monolithic architecture. In future work, we must determine how to integrate legacy applications into the context choreography and context map.

B. Conclusion

During our research, we found many different studies that consider model-driven approaches for developing microservices. Using these approaches for microservices is common. In domain-driven design, especially, the approaches that focus on the development itself but omit the design and maintenance phases. Therefore, we wanted to provide details on the design and maintenance of a microservice architecture using DDD's context map.

The context map has great potential to aid in developing and maintaining applications and is more useful when considering a microservice architecture. However, the context map shares a problem with most DDD concepts: its lacking placement in software engineering, foundations and concrete guidelines. Therefore, we first provided placement for the context map. Next, we clarified its foundations with a focus

on the bounded context, the main concept of the context map. It needs to be emphasized that the context map is not only showing microservices and therefore the microservice architecture. Communication paths and dependencies between development teams are also considered in the context map. After the foundations were clear, we could develop a systematic approach for creating the context map. This approach began in the analysis phase with an initial domain model, separating the domain knowledge into bounded contexts, stating relationships between them, and putting the bounded contexts into a context map. The separation of bounded contexts and their relationships are stated in our new diagram, the "context choreography." This diagram's purpose is to illustrate necessary bounded contexts for microservice-based applications. Furthermore, we created UML profiles for the context map and context choreography to unify their appearances. With those profiles, we especially tackle the various interpretations of the context map and support the inter-team communication.

This article's contributions are the first step in making use of the context map, and the newly defined context choreography more efficient.

REFERENCES

[1] B. Hippchen, M. Schneider, I. Landerer, P. Giessler, and S. Abeck, "Methodology for splitting business capabilities into a microservice architecture: Design and maintenance using a domain-driven approach," *The Fifth International Conference on Advances and Trends in Software Engineering*, 2019.

[2] M. Gebhart, P. Giessler, and S. Abeck, "Challenges of the Digital Transformation in Software Engineering," *ICSEA 2016*, p. 149, 2016.

[3] C. Richardson, "Pattern: Messaging," https://microservices.io/patterns/communication-style/messaging.html [retrieved: 11, 2019].

[4] E. Evans, *Domain-Driven Design: Tackling Complexity in the Heart of Software*. Addison-Wesley Professional, 2004.

[5] V. Vernon, Ed., *Implementing Domain-Driven Design*. Addison-Wesley, 2013.

[6] S. Newman, *Building Microservices: Designing Fine-grained Systems*. " O'Reilly Media, Inc.", 2015.

[7] M. E. Conway, "How do Committees Invent," *Datamation*, vol. 14, no. 4, pp. 28–31, 1968.

[8] M. Fowler, "HumaneRegistry," URL: https://martinfowler.com/bliki/HumaneRegistry.html [retrieved: 08, 2019].

[9] Google, "apigee, API management," https://apigee.com/api-management/ [retrieved: 08, 2019].

[10] B. Hippchen, P. Giessler, R. Steinegger, M. Schneider, and S. Abeck, "Designing Microservice-Based Applications by Using a Domain-Driven Design Approach," in *International Journal on Advances in Software, Vol. 10, No. 3&4, pp. 432–445*, 2017.

[11] M. Bruce and P. A. Pereira, *Microservices in Action*. Manning Publications, 2019.

[12] Pivotal, "Rabbitmq," https://www.rabbitmq.com [retrieved: 11, 2019].

[13] Apache, "Kafka," https://kafka.apache.org [retrieved: 11, 2019].

[14] B. Bruegge and A. H. Dutoit, *Object-Oriented Software Engineering Using UML, Patterns and Java-(Required)*. Prentice Hall, 2004.

[15] O. Vogel, I. Arnold, A. Chughtai, and T. Kehrer, *Software Architecture: A Comprehensive Framework and Guide for Practitioners*. Springer Science & Business Media, 2011.

[16] S. Millett, *Patterns, Principles and Practices of Domain-Driven Design*. John Wiley & Sons, 2015.

[17] L. Bass, I. Weber, and L. Zhu, *DevOps: A Software Architect's Perspective*. Addison-Wesley Professional, 2015.

[18] L. Chen, "Continuous delivery: Overcoming adoption challenges," *Journal of Systems and Software*, vol. 128, pp. 72–86, 2017.

[19] A. Balalaie, A. Heydarnoori, and P. Jamshidi, "Microservices architecture enables devops: Migration to a cloud-native architecture," *IEEE Software*, vol. 33, no. 3, pp. 42–52, 2016.

[20] L. Chen, "Microservices: Architecting for continuous delivery and devops," in *2018 IEEE International Conference on Software Architecture (ICSA)*. IEEE, 2018, pp. 39–397.

[21] J. F. Smart, *BDD in Action: Behavior-Driven Development for the Whole Software Lifecycle*. Manning, 2015.

[22] E. Evans, *Domain-Driven Design Reference: Definitions and Pattern Summaries*. Dog Ear Publishing, 2014.

[23] J. Osis and U. Donins, *Topological UML Modeling: An Improved Approach for Domain Modeling and Software Development*, ser. Computer Science Reviews and Trends. Elsevier Science, 2017.

[24] O. OMG, "Unified Modeling Language (OMG UML)," *Superstructure*, 2007.

[25] Z. Durdik, *Architectural Design Decision Documentation Through Reuse of Design Patterns*. KIT Scientific Publishing, 2016, vol. 14.

[26] W. F. Tichy, "Hints for reviewing empirical work in software engineering," *Empirical Software Engineering*, vol. 5, no. 4, pp. 309–312, 2000.

[27] K. Hightower, B. Burns, and J. Beda, *Kubernetes: Up and Running: Dive Into the Future of Infrastructure*. O'Reilly Media, 2017.

[28] M. Schneider, B. Hippchen, P. Giessler, C. Irrgang, and S. Abeck, "Microservice development based on tool-supported domain modeling," *The Fifth International Conference on Advances and Trends in Software Engineering*, 2019.

[29] I. Nadareishvili, R. Mitra, M. McLarty, and M. Amundsen, *Microservice Architecture: Aligning Principles, Practices, and Culture*. " O'Reilly Media, Inc.", 2016.

[30] A. Brandolini, "Introducing Event Storming," *blog, Ziobrando's Lair*, vol. 18, 2013, URL: http://ziobrando.blogspot.com/2013/11/introducing-event-storming.html [retrieved: 0, 2019].

[31] S. Tyszberowicz, R. Heinrich, B. Liu, and Z. Liu, "Identifying Microservices Using Functional Decomposition," pp. 50–65, 2018.

[32] N. Dmitry and S.-S. Manfred, "On Micro-Services Architecture," *International Journal of Open Information Technologies*, vol. 2, no. 9, 2014.

[33] M. Wynne, A. Hellesoy, and S. Tooke, *The Cucumber Book: Behaviour-Driven Development for Testers and Developers*. Pragmatic Bookshelf, 2017.

[34] F. Rademacher, S. Sachweh, and A. Zündorf, "Towards a uml profile for domain-driven design of microservice architectures," in *International Conference on Software Engineering and Formal Methods*. Springer, 2017, pp. 230–245.

A Controller for Anomaly Detection, Analysis and Management for Self-Adaptive Container Clusters

Areeg Samir, Nabil El Ioini, Ilenia Fronza, Hamid R. Barzegar, Van Thanh Le and Claus Pahl

Faculty of Computer Science
Free University of Bozen-Bolzano
39100 Bolzano, Italy
Email: `firstname.surname@unibz.it`

Abstract—**Service computing in the cloud allows applications to be deployed remotely. These are managed by third-party service providers that make virtualised resources available for these services. Self-adaptive features for load-balancing and auto-scaling are available here, but generally there is no direct access to the infrastructure or platform-level execution environment. Some quality parameters of a provided service can be directly observed while others remain hidden from the service consumer. Our solution is an autonomous self-adaptive controller for anomaly remediation in this semi-hidden setting. The objective of the controller is to, firstly, determine possible root causes of consumer-observed anomalies and, secondly, take appropriate action. This needs to happen in an underlying provider-controlled infrastructure. We use Hidden Markov Models to map observed performance anomalies into hidden resources, and to identify the root causes of the observed anomalies. We apply the model to a clustered computing resource environment that is based on three layers of aggregated resources. We discuss use cases to illustrate the utility of the proposed solution.**

Index Terms—*Cloud Computing; Container Technology, Distributed Clusters; Markov Model; Anomaly Detection; Anomaly Analysis; Workload; Performance.*

I. INTRODUCTION

Due to the dynamic nature of loads in a distributed cloud or edge computing setting, consumers may experience anomalies (e.g., variation in a resource performance) due to distribution, heterogeneity, or scale of computing that may lead to performance degradation and potential application failures. Furthermore, loads might vary over time:

- changes of the load on individual resources,
- changing workload demand and prioritisation,
- reallocation or removal of resources in dynamic environments.

These may affect the workload of current system components (container, node, cluster), and may require rebalancing their workloads. These cloud and edge computing settings allow services to be provided by allowing applications to be deployed and managed by third-party providers. These make shared virtualised resources accompanied by dynamic management facilities [1],[2],[3],[4] available.

Recent works such as [5],[6],[7] have looked at resource usage, rejuvenation, or analysing the correlation between resource consumption and abnormal behaviour of applications.

Less attention has been given to the possibly hidden reason behind the occurrence of an observable performance degradation (root cause) [8], and how to deal with the degradation in a hierarchically organised cluster setting.

In order to address these challenges in a shared virtualised environment, third party providers provide some factors that can be directly observed (e.g., the response time of service activations) while others remain hidden from the consumer (e.g., the reason behind the workload, the possibility to predict the future load behaviour, the dependency between the affected nodes and their loads in a cluster).

The core solution presented in this paper is a controller [9],[10] that automatically detects the anomalous behaviour within a cluster of containers running on cluster nodes, where a sequence of observations is emitted by the system resource. The controller remedies the detected anomalies that occur at the container, node or cluster level. To achieve that this paper: (i) analyse the possible causes of observable anomalies in an underlying provider-controlled infrastructure; (ii) define an anomaly detection, and analysis controller for a self-adaptive cluster environment, that automatically manages the resource workload fluctuations. Our objective is to introduce this controller in terms of its architecture and processing activities. We expand our earlier work in [1],[18] here by discussing context and the application of the architecture in use cases in more detail.

A specific feature of our solution is the differentiation between two types of observations:

- System states (anomaly/fault) that refer to anomalous or faulty behaviour, which is hidden from the consumer. This indicates that the behaviour of a system resource is significantly different from normal behaviour. An anomaly in our case may point to an undesirable behaviour of a resource such as overload, or to a desirable behaviour like underload of a system resource, which can be used as a solution to reduce the load at overloaded resources.
- Emission or Observation (observed failure from these states), which indicates the occurrence of failure resulting from a hidden state.

We focus on technical concerns in relation to workload and response time fluctuations.

To address this problem, we use so-called Hierarchical Hidden Markov Models (HHMMs) [11] as a stochastic model to map the observed failure behaviour of a system resource to its hidden anomaly causes (e.g., overload) through tracking the detected anomaly to locate its root cause. We implement the proposed controller for a clustered computing resource environment targeting specifically container technologies.

The paper is structured as follows. Section II reviews related work extensively. Section III discusses use cases to clarify the problem context and how an anomaly solution can address the problems. Section IV explains the motivation behind our work in terms of the proposed architecture. Section V gives a brief introduction of HHMM as the central formal construct. Section VI explains the mapping of failure and fault. Section VII explains the controller architecture with its analysis and recovery components. Section VIII evaluates the proposed architecture. As an outlook into the transferability of the architecture, we discuss trust anomalies in Section IX. Section X concludes the paper.

II. RELATED WORK

There are a number of studies that have addressed workload analysis in dynamic environments [5],[6],[7]. They proposed various methods for analysing and modelling workload.

Dullmann [17] provides an online performance anomaly detection approach that detects anomalies in performance data based on discrete time series analysis. Peiris et al. [21] analyse root causes of performance anomalies by combining the correlation and comparative analysis techniques in distributed environments. Sorkunlu et al. [22] identify system performance anomalies through analysing the correlations in the resource usage data. Wang et al. [7] propose to model the correlation between workload and the resource utilization of applications to characterize the system status. Maurya and Ahmad [24] propose an algorithm that dynamically estimates the load of each node and migrates the task if necessary. The algorithm migrates the jobs from overloaded nodes to underloaded ones through working on pair of nodes, it uses a server node as a hub to transfer the load information in the network, which may result in overhead at the node.

Moreover, many works have used the HMM and its derivations to detect anomaly. In [25], the author proposes various techniques implemented for the detection of anomalies and intrusions in the network using the HMM. In [27] the author detects faults in real-time embedded systems using the HMM through describing the healthy and faulty states of a system's hardware components. In [29], HMM is used to find, which anomaly is part of the same anomaly injection scenarios.

A fault tolerance management solution for physical and virtual machines level is presented in [34]. It uses Redundant Array of Independent Disks technology in order to optimize storage space and to recover data in case of failure. More concretely, the author divides a set of VM and PM into subsets of the same size. Then, two services are used to collect information about a resource status and to manage resources through adding and removing resources in order to mitigate resource failure. Here, only two aspects of recovery handling, namely handling storage disk crash and system crashes are considered. In [33], the detection of anomalous behaviours (such as CPU overload or Denial of Service Attacks) are the topic. The authors provide an adaptation policy based on a multi-dimensional utility-based model. The score and likelihood for the anomaly detected to select adaptation policy is provided in order to scale compute resources. Here, a node leader for each microservice cluster is selected and each node maintains the cluster state and preserves the cluster records. This node leader also decides on the adaptation policy action. In this work, however, only two types of anomalies are handled. It also limits actions to mitigate the anomalous behaviour to horizontal and vertical auto-scaling actions. Furthermore, prediction is not included.

In [35], performance and prediction are the key concerns. The performance of several machine learning models is investigated in order to predict attacks on the IoT systems accurately. A random forest technique is shown to achieve good anomaly prediction in comparison to other machine learning solutions. Nonetheless, the focus is on predicting network anomaly only. In [36], a solution to estimate the capacity of a microservice is presented. Here measuring the maximal number of successfully processed user requests per second for a given service such that no SLO is violated is the key idea. The authors carry out a number of load tests and then fit an appropriate regression model to the acquired performance data. This work investigates the impact of workload on measures CPU and memory usage. Changing the number of requests affects the number of virtual CPU cores does not affect the memory utilization significantly. What is somewhat neglected is the dependency between nodes and services. Predicting future workload is not covered.

Localizing faulty resources in cloud environments through modelling correlations among anomalous resources is addressed in [65]. Graph theory is employed in order to locate the correlation between pairs of resources. analysing the amount of occupied memory in a physical server, the CPU consumption of a virtual machine and the number of connections accepted by an application is the focus. This work does not cover anomalies specifically in microservice or container environments. In [39] the authors focus on detecting anomalous behaviour of services deployed on VM in cloud environment. Like our architecture, different anomaly injection scenarios are created and workload is generated to test the impact of anomaly on the cloud services. The authors emulated different anomalies at the CPU, memory, disk, and network. However, their work does not track the cause of anomalous behaviour in containerized cluster environment, and it neglects the dependency between nodes.

More dedicated to microservices and containers are:

- In [69], the authors investigate the mutual impact of microservices on the same host. This study looks at the

consequences of these side effects have on failure prediction. For this, the authors measure the CPU, memory and network usage metrics of the containers and nodes. The work evaluates the current failure prediction methods in a microservice architecture, but does not locate or detect anomalous behaviour. The work focus is CPU-bound workload.

- Kratzke [43] looks at the impact of network performance on containers deployed on VMs. He carries out a number of experiments in order to analyse the network performance of containers by using horizontal scaling and considering the network data transfer rate. Nonetheless, the focus is on network aspects and their impact on container performance.

Another couple of studies focus on performance:

- Wert [47] presents a specification language, called the Performance Problem Diagnostics Description Model, in order to specify information needed for an automatic performance problem diagnostics. Here, workload is analysed to detect performance faults and categorize performance faults into three layers: (i) symptoms, which are externally visible indicators of a performance problem, (ii) manifestation, which are internal performance indicators, and (iii) finally root-causes, which are the physical factors whose removal eliminates the manifestations and symptoms of a performance problem. The author's approach does not consider dependencies between faults nor avoids human interaction. For example, there should be heuristics to be able to detect performance problems. The approach is designed to apply for a specific application domain. A recovery mechanism for the detected faults is not covered. Also, dependencies between anomalies are not addressed. The solution is based on predefined heuristics (rules) in order to detect performance problems. Consequently, applying the approach on a different domain or changing the fault model requires heuristics update.
- Ibidunmoye et al. [72] look at the detection the anomalous behaviour in performance using forecasting model to estimate the bandwidth, detect performance changes and decompose time series into components. In this work, hard thresholds are used in all datasets, which might not reflect the actual workloads in system accurately. They only look at the detection of anomalies without a further analysis. Labelled-time is used there, which is often not suitable in order to detect all anomalies as some anomalies could not be discovered during the detection process and time complexity in terms of data size may occur.

Prediction is another important concern:

- Predicting the impact of processor cache interference within consolidated workloads is the focus of [62]. In order to predict the performance degradation of these consolidated applications, the proposed prediction solution is linear in terms of the number of cores sharing the last-level cache. The authors limit their discussion to cache contention issues, ignoring other resource types and resulting faults.
- Guan and co-authors have implemented a probabilistic prediction model using a supervised learning method in [48]. This model serves to detect anomalous behaviour in cloud-based environments by analysing the correlation between different selected metrics (including CPU, memory, disk, and network concerns) in order to determine essential metrics that characterize the correlation between performance and an anomaly event. Here, directed acyclic graphs DAGs are used in order to analyse the correlation of the different performance metrics with failure events in both virtual and physical machines. The authors determine in the paper the conditional probability of each metric for the anomaly occurrences. Those metrics where conditional probabilities are greater than a predefined threshold are then selected. Nevertheless, their results show that their model suffers from poor prediction efficiency when it is used to predict cloud anomalies.
- In [67], the authors introduce a general-purpose prediction model that aims at preventing anomalies in cloud environment. Their supervised learning-based model utilised as in our case Markov models. There they combine two dependent Markov chains with a tree augmented the Bayesian network. Statistical learning algorithms are applied based on system-level metrics (CPU, memory, network I/O statistics) aiming to predict anomalous behaviour. A limitation is that the authors do not discuss prediction efficiency.

Nathuji et al. [78] look at a control theoretic consolidation solution that aims at mitigating effects of anomalies in the context of cache, memory and hardware contention of co-existing workloads. Their solution manages the interference between consolidated virtual machines by dynamically adapting resource allocations to applications based on workload SLAs. The focus in that paper is on CPU-bound workload and compute-intensive applications. Monitoring is at the centre of [77]. They discuss a technique for localizing anomalies at runtime using the Kieker monitoring approach. For anomaly localization, an anomaly score is calculated for each operation using a specified threshold. A set of rules is given to detect performance anomaly, which are continuously evaluated based on the anomaly score by utilising forecasting techniques to predict future values in time series. Experimentally observed measurement values such as response times are analysed with the forecasted values to detect anomalies. Different types of performance anomalies and anomaly dependencies are however not considered.

The objective of this paper here is to detect and locate the anomalous behaviour in containerized cluster environment [28] through considering the influence of dynamic workloads on their anomaly detection solutions. The proposed controller consists of:

- (1) *Monitoring*, that collects the performance data of

(services, containers, nodes 'VM') such as CPU, memory, and network metrics;

- (2) *Detection*, that detects anomalous behaviour, which is observed in response time of a component;
- (3) *Identification*, which tracks the cause of the detected anomaly.
- (4) *Recovery*, that heals the identified anomalous components.
- (5) *Anomaly injection*, which simulates different anomalies, and gathers dataset of performance data representing normal and abnormal conditions.

III. A DISCUSSION OF USE CASES

In order to better motivate our solution, we look at two use cases now. One looks at a widely used cloud-based scenario. Here, we assume a cluster of containers that are managed by an orchestrator such as Kubernetes or Docker Swarm. The other use case considers an edge cloud setting, where again a cluster architecture, but this time deployed on constrained hardware devices to host the container cluster is considered.

A. Use Case 1: Cloud-centric Container Orchestration

Container technology is increasingly popular recently. It is now widely used as the mechanism for software deployment. Containers as a more lightweight form of virtualisation compared to virtual machines (VMs) consume less resources. They compare favourably to VMs in terms of startup time to memory/storage needs. This applies also to cloud environments. Many cloud infrastructure (IaaS) providers and platform service (PaaS) providers provide different container deployment solutions. In many of these cases, an orchestration tool like Kubernetes[1], see Figure 1, or Docker Swarm, is used to support the automated deployment, scaling and management of containerized applications are used by the providers, see Figures 2 and 3. These are typically homogeneous cloud container cluster in terms of the underlying infrastructure.

This setting, however, creates problems regarding monitoring and problem detection. A concrete problem that becomes obvious here is that a service consumer generally have access to monitoring data at an (application or platform) service level, but not necessarily at the underlying physical infrastructure level [44], which is hidden by the service provider. Nonetheless, service consumer are often given access to controllers that can for instance auto-scale the application deployed.

In this case, our solution can be applied. The user could be provided with a anomaly management architecture (essentially a trained Hidden Markov Model HMM, as we will see later). This model then reflects possible underlying (and unobservable) faults for the failures that have been observed by the service consumer.

[1]https://kubernetes.io/
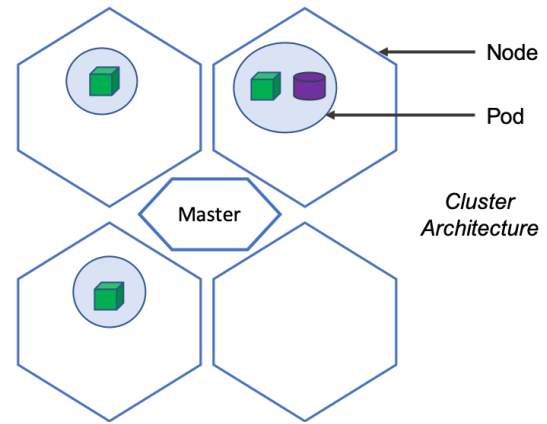
FIGURE 1. A CONTAINER CLUSTER ARCHITECTURE BASED ON KUBERNETES.
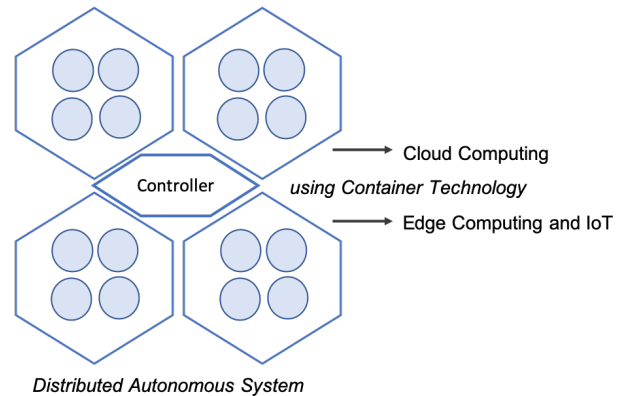
FIGURE 2. A DISTRIBUTED CONTAINER SYSTEM SUITABLE FOR CLOUD AND EDGE COMPUTING.

B. Use Case 2: Edge Cloud Orchestration – Connected Cars

The lightweightness of containers as introduced above makes them very suitable to be utilised outside a classical centralised cloud setting. Here, edge cloud infrastructures that provide computational capabilities for IoT or other remote application can benefit from the containers' lightweightness. This is in particular useful if the edge infrastructure is limited in terms of its capabilities. For this type of situation, we assume now a cluster on single-board devices as the
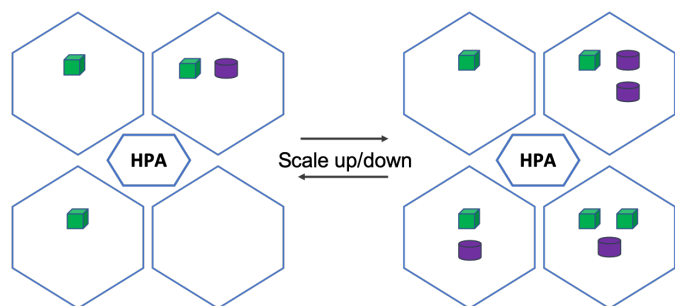
FIGURE 3. AUTO-SCALING WITH KUBERNETES – THE HORIZONTAL POD AUTOSCALER (HPA).
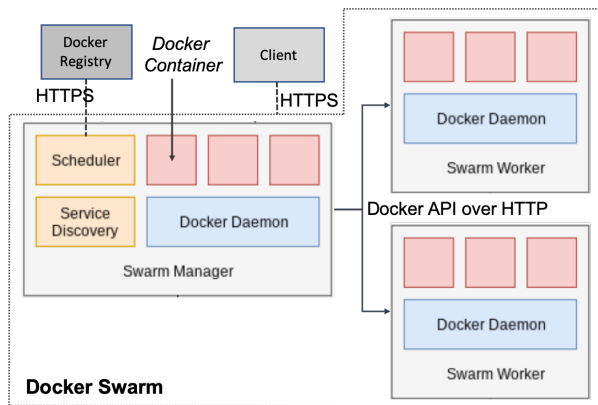
FIGURE 4. CONTAINER ORCHESTRATION WITH DOCKER SWARM.

FIGURE 5. A DOCKER ARCHITECTURE FOR DATA STREAM PROCESSING HOSTED ON RASPBERRY PI (RPI) DEVICES.

physical infrastructure to host the container cluster platform. Specifically, we use single-board Raspberry Pi[2] devices. In our experiments, we use Docker Swarm[3] as the container orchestration tool, see Figure 4.

We first introduce the general edge cluster architecture and then illustrate this through a connected car use case.

Our proposed solution is based on a categorisation of hidden fault and observable failure cases, in which observable failures (to meet QoS requirements of the consumer) are mapped to their root causes, i.e., the underlying hidden faults that have caused them. Examples are an overloaded container that simply slows down or a neighbouring container on the same node on which a concrete container depends (e.g., is waiting for an answer for a request) [18],[19]. We use Markov models in our formalisation that reflect the possibility of several causes and the likelihood of each of these. Typical (hidden) fault types are CPU hog, network latency or workload contention, while only response time failures are observable. For each of these mapping cases, we have associated suitable remedial actions, such as workload distribution, container migration or resource rescaling.

A technical term for the connected cars scenario is CCAM. CCAM stands for connected Cooperative, Connected and Automated Mobility. We specifically look at connected cars in this context to make the problem more concrete. CCAM brings in this context the world of 5G telecommunications, automotive solutions and cloud and edge computing together.

Concrete use cases are car manoeuvre support, for instance for lane changing, or video streaming. For lane changing, several cars might need to be coordinated in their actions using mobile edge clouds. Here, latency is a critical issue and needs to be constantly monitored.

There are a number of identifiable problems:

- node overload: both on-board units as well as road-side units in a connected car situation are very limited in terms of their computational and storage capacities.

---

[2]https://www.raspberrypi.org/
[3]https://docs.docker.com/engine/swarm/
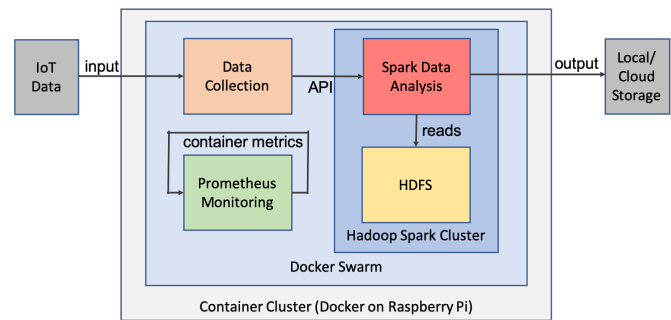
- connection dependency: 5G is normally considered as the minimum required standard in order to meet the required latency needs. Even with 5G enabled, a high density of cars combined with the need to support while the cars are moving creates capacity challenges also for the network.

Solutions that are applicable as remedial actions are:

- allocate more capacity, e.g., for video buffering
- migrate resources to avoid network problems (migrate container and/or data/state)
- repurpose nodes (redeploy other containers)

Our work in [20],[46] demonstrates how a container cluster solution implemented on Raspberry Pis can support this type of scenario. There, a Docker Swarm based management supports containers for data stream processing (Apache Spark), supported by Prometheus as a monitoring tool, Grafana for analyse/visualised data and databases like InfluxDB to store data, see Figures 5 and 6.

A concrete technology that involves mobile edge clouds and additional on-board and road-side assistance is CCAM – Cooperative, Connected and Automated Mobility.

Both node and connection problem can trigger an anomaly management system. Once an anomaly is identified, actions such as moving containers to avoid the node/connection problem or repurpose nodes to meet changing needs, indicated for instance by underload, are pursued.

In this context, the decision model can be a Hidden Markov Model HMM. The probabilities reflected in the HMM represent the following aspects:

- the adequacy of the failure/fault mapping (identify anomaly),
- the suitability of the recovery action (recover anomaly).

The HMM identifies different anomaly states [1]. These are dependent on the monitored performance and workload/utilisation metrics. In other works [9],[10], we have used fuzzy logic to map monitored data to so-called membership functions that represent these different states. We refer the reader to these works for more detail. Here, we focus on the anomaly processing.
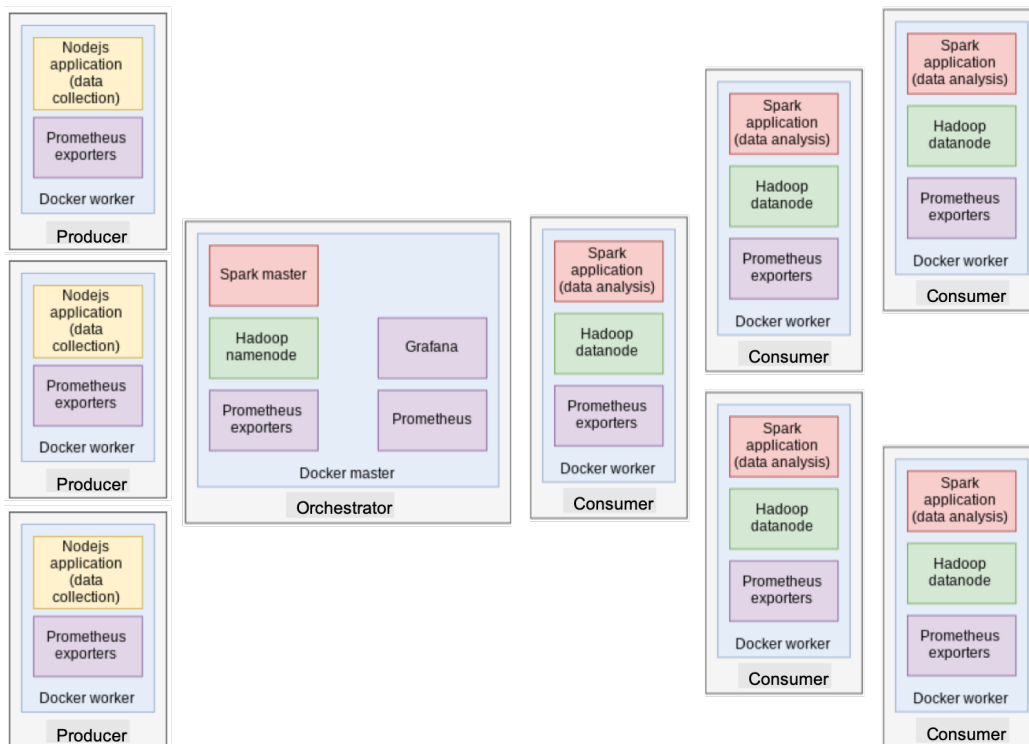
FIGURE 6. A DOCKER CONTAINER DISTRIBUTION FOR THE DATA STREAM PROCESSING APPLICATION ON RPIS.

IV. MOTIVATING EXAMPLE FOR THE CONTROLLER ARCHITECTURE

A failure is the inability of a component to perform its functions with respect to a specified (e.g., performance) requirements [26]. Faults (also called anomalies) are system properties that describe an exceptional condition occurring in the system operation that may cause one or more failures [31].

We assume that a failure is a kind of unexpected response time observed during system component runtime (i.e., observation), while fluctuations occurring during a resource execution of a component are considered as faults or anomalies (state of a hidden component). For example, fluctuations in workload such as overload faults may cause delay in a system response time (observed failure).

Generally, the observed metrics do not provide enough information to identify the cause of an observed failure. For example, the CPU utilization of a containerized application is about 30% with 400 users, and it increases to about 70% with 800 users in the normal situation. Obviously, the system is normal with 800 users. But probably the system shows anomalous behaviour with 400 users, when the CPU utilization is about 70%. Thus, it is hard to identify whether the system is normal or anomaly just based on the CPU utilization. Thus, specifying a threshold for the utilization of resource without considering the number of users, will raise anomalous behaviours. Consequently, it is important to integrate the data of workload into anomaly detection and identification solutions.

Once provided with a link between faults (workloads), and failures (response time) emitted from components, we can also apply a suitable recovery strategy depending on the type of identified fault.

Thus, a self-adaptation controller will be introduced later in this paper to automatically manage faults through identifying the degradation of performance, determining the dependency between faults and failures, and applying recovery strategies. We can align the steps of the fault management with the Monitoring, Analysis, Planning, Execution, and Knowledge (MAPE-K) control loop as a conceptual framework.

V. HIERARCHICAL HIDDEN MARKOV MODEL (HHMM)

The Hierarchical Hidden Markov Model (HHMM) [11] is a generalization of the Hidden Markov Model (HMM) that is used to model domains with hierarchical structure (e.g., intrusion detection, plan recognition, visual action recognition). The HHMM can characterize the dependency of the workload (e.g., when at least one of the states is heavy loaded). The states (cluster, node, container) in the HHMM are hidden from the observer and only the observation space is visible (response time). The states of HHMM emit sequences rather than a single observation by a recursive activation of one of the substates (nodes) of a state (cluster). This substate might also be hierarchically composed of substates (containers). Each container has an application that runs on it. In case a node or a container emits observation, it will be considered a production state. The states that do not emit observations directly are called internal states. The activation of a substate

by an internal state is a vertical transition that reflects the dependency between states. The states at the same level have horizontal transitions. Once the transition reaches to the End state, the control returns to the root state of the chain as shown in Figure 7. The edge direction indicates the dependency between states.

We choose the HHMM as every state can be represented as a multi-level HMM in order to: (1) show communication between nodes and containers, (2) demonstrate the impact of workloads on the resources, (3) track the anomaly cause, and (4) represent the response time variations that emit from nodes or containers.

VI. FAILURE-TO-FAULT MAPPING

Based on analysing the log file and monitored metrics from existing systems, we can obtain knowledge regarding (1) the dependencies between containers, nodes and clusters; (2) response time fluctuations emitted from containers or nodes; (3) workload fluctuations that cause changes in response time. We need a mechanism that automatically maps a type of anomaly to its causes. We can identify different failure-fault cases that may occur at container, node or cluster level as illustrated in Figure 8. We focus on addressing the correlation between workload (overload) and the response time at container, node, and cluster.

A. Low Response Time Observed at Container Level

There are different reasons that may cause this:

- *Case 1.1. Container overload (self-dependency)*: means that a container is busy, causing low response times, e.g., c_1 in N_1 has entered into load loop as it tries to execute its processes while N_1 keeps sending requests to it, ignoring its limited capacity.
- *Case 1.2. Container sibling overloaded (internal container dependency)*: this indicates another container c_2 in N_1 is overloaded. This overloaded container indirectly affects the other container c_1 as there is a communication between them. For example, c_2 has an application that almost consumes all resources. The container has a communication with c_1. At such situation, when c_2 is overloaded, c_1 will go into underload. The reason is that c_2 and c_1 share the resources of the same node.
- *Case 1.3. Container neighbour overload (external container dependency)*: this happens when a container c_3 in N_2 is linked to another container c_2 in another node N_1. In another case, some containers c_3 and c_4 in N_2 dependent on each other, and container c_2 in N_1 depends on c_3. In both cases c_2 in N_1 is badly affected once c_3 or c_4 in N_2 are heavily loaded. This results in low response time observed from those containers.

B. Low Response Time Observed at Node Level

There are different reasons that cause such observations:

- *Case 2.1. Node overload (self-dependency)*: generally, node overload happens when a node has low capacity, many jobs waited to be processed, or problem in network. Example, N_2 has entered into self-load due to its limited capacity, which causes an overload at the container level as well c_3 and c_4.
- *Case 2.2. External node dependency*: occurs when low response time is observed at node neighbour level, e.g., when N_2 is overloaded due to low capacity or network problem, and N_1 depends on N_2. Such overload may cause low response time observed at the node level, which slow the whole operation of a cluster because of the communication between the two nodes. The reason behind that is N_1 and N_2 share the resources of the same cluster. Thus, when N_1 shows a heavier load, it would affect the performance of N_2.

C. Low Response Time Observed at Cluster Level

If a cluster coordinates between all nodes and containers, we may observe low response time at container and node levels that cause difficulty at the whole cluster level, e.g., nodes disconnected or insufficient resources.

- *Case 3.1. Communication disconnection* may happen due to problem in the node configuration, e.g., when a node in the cluster is stopped or disconnected due to failure or a user disconnect.
- *Case 3.2. Resource limitation* happens if we create a cluster with too low capacity which causing low response time observed at the system level.

The mapping between faults and failures needs to be formalised in a model that distinguishes observations and hidden states. Thus, HHMM is used to reflect the system topology.

VII. SELF-ADAPTIVE CONTROLLER ARCHITECTURE

This section explains the controller architecture (Figure 9).

A. Managed Component Pool

The system under observation consists of a cluster that is composed of a set of nodes that host containers as the application components. A node could be a virtual machine that has a given capacity. The main job of the node is to assign requests to its containers. Containers are stand-alone, executable packages of software. Multiple containers can run on the same node, and share the operating environment with other containers. Each component either cluster, node, or container may emit observations. Observations are emissions of failure from a component resource.

We install an agent on each node to collect metrics from the pool, and to expose log files of containers and nodes to Real-Time/Historical Data storage. The agent adds data interval function to determine the time interval at which the data collected belongs. The data interval function specifies the lower and upper limits for the data arrivals. The response time,
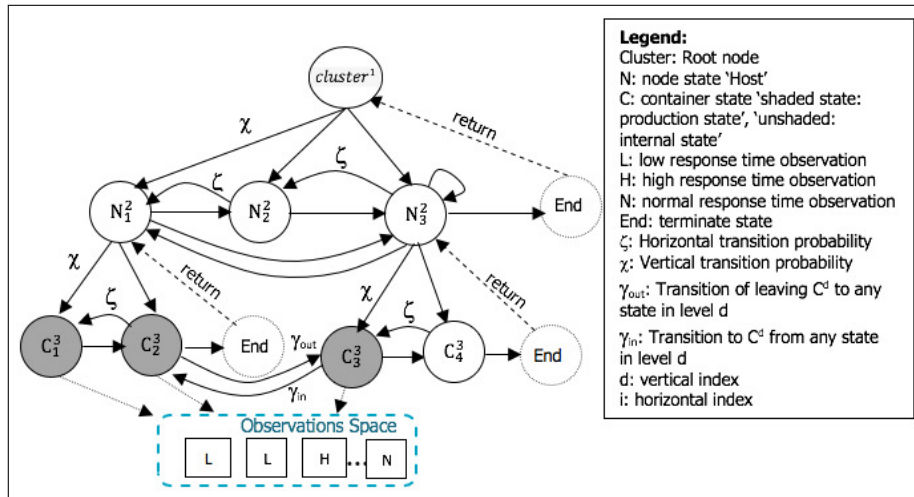
FIGURE 7. THE HHMM FOR WORKLOAD.
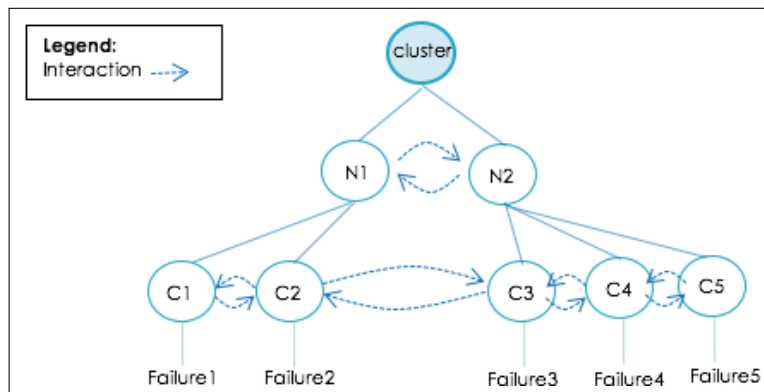
FIGURE 8. DEPENDENCIES BETWEEN CLUSTER, NODES AND CONTAINERS.
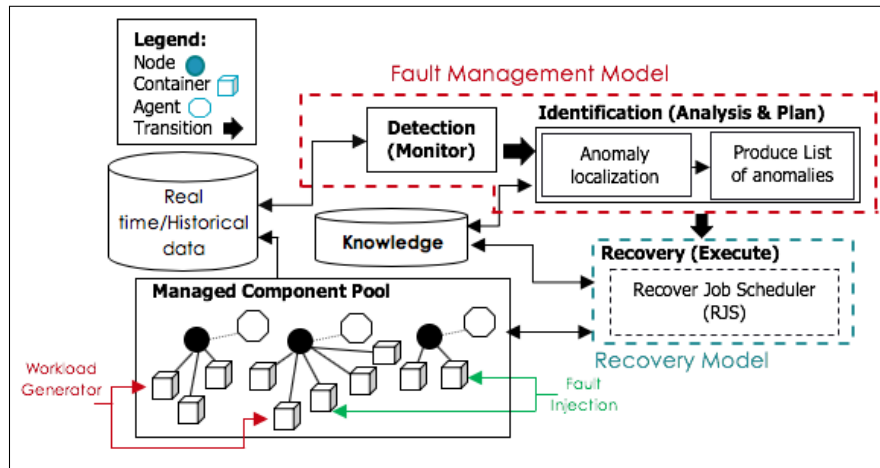
FIGURE 9. THE ANOMALY MANAGEMENT CONTROLLER ARCHITECTURE.

and the state of the component are assigned to each interval. Moreover, the agent gathers data regarding the workload (i.e., no. of requests issued to component), and monitored metrics (i.e., CPU, Memory) to characterize the workload of components processed in an interval. The agent pushes the data to be stored in the Real time/Historical storage to be used by the Fault Management Model.

B. Fault Management Model

The model is based on the history of the overall system performance. This can be used to compare the predicted status with the currently observed one to detect anomalous behaviour. The fault management model consists of:

a) Detection (Monitor): To detect anomaly, the monitor collects system data from the Real time/Historical storage. Then, it checks if there is anomalous behaviour at the managed components through utilizing spearmans rank correlation coefficient to estimate the dissociation between the response time and the number of requests (workload). If there is a decrease in the correlation degree, then the metric is not associated with the increasing workload, which means the observed variation in performance is not an anomaly. In case the correlation degree increase, this refers to the existence of anomaly occurred as the impact of dissociation between the workload and the response time exceeds a certain value. To achieve that we wrote an algorithm to be used as a general threshold to highlight the occurrence of anomaly in the managed pool under different workloads. We added a unique workload identifier to the group of workloads in the same period to achieve traceability through the entire system. We specified that the degree of dissociation (DD = 15) can be used as an indicator for performance degradation considering different response time, and different workloads. The value of DD will be compared against the monitored metrics (i.e., CPU, Memory utilization) to detect anomalous behaviour within the system. In case an anomaly is detected, the controller moves to the fault management to track the cause of anomaly in the system.

b) Identification (Analysis and Plan): Once there is appearance of anomaly, we build HHMMs to identify anomalies in system components as shown in Figure 7.

The HHMM vertically calls one of its substates $N_1^2 = \{C_1^3, C_2^3\}$, N_2^2, $N_3^2 = \{C_3^3, C_4^3\}$ with vertical transition χ and d index (superscript), where $d = \{1, 2, 3\}$. Since N_1^2 is abstract state, it enters its child HMM substates C_1^3 and C_2^3. Since C_2^3 is a production state, it emits observations, and may make horizontal transition γ, with i horizontal index (subscript), where $i = \{1, 2, 3, 4\}$, from C_1^3 to C_4^3. Once there is no another transition, C_2^3 transits to the end state End, which ends the transition for this substate, to return the control to the calling state N_1^2. Once the control returns to the state N_1^2, it makes a horizontal transition (if exist) to state N_2^2, which horizontally transits to state N_3^2. State N_3^2 has substates C_3^3 that transits to C_4^3 which may transit back to C_3^3 or transits to the End state. Once all the transitions under this node are achieved, the control returns to N_3^2. State N_3^2 may loop around, transits back to N_3^2, or enters its End state, which ends the whole process and returns control to the cluster. The model cannot horizontally do transition unless it vertically transited. Further, the internal states do not need to have the same number of substates. It can be seen that N_1^2 calls containers C_1^3 and C_2^3, while N_2^2 has no substates. The horizontal transition between containers reflect the request/reply between the client/server

in our system under test, and the vertical transition refers to child/parent relationship between containers/node.

The observation O is denoted by $F_i = \{f_1, f_2, ..., f_n\}$ to refer to the response time observations sequence (failures). An observed low response time might reflect workload fluctuation. This fluctuation in workload is associated with a probability that reflects the state transition status from OL to NL ($PF_{OL \to NL}$) at a failure rate $\Re$, which indicates the number of failures for a N, C or $cluster$ over a period of time.

We use the generalized Baum-Welch algorithm [11] to train the model by calculating the probabilities of the model parameters: (1) the horizontal transitions from a state to another. (2) probability that the O is started to be emitted for $state_i^d$ at t. $state_i^d$ refers to container, node, or cluster. (3) the O of $state_i^d$ are emitted and finished at t. (4) the probability that $state^{d-1}$ is entered at t before O_t to activate state $state_i^d$. (5) the forward and backward transition from bottom-up.

The output of algorithm is used to train Viterbi algorithm to find the anomalous hierarchy of the detected anomalous states. As shown in "(1)-(3)", we recursively calculate $\Im$ which is the ψ for a time set ($\bar{t} = \psi(t, t+k, C_i^d, C^{d-1})$), where ψ is a state list, which is the index of the most probable production state to be activated by C^{d-1} before activating C_i^d. $\bar{t}$ is the time when C_i^d is activated by C^{d-1}. The δ is the likelihood of the most probable state sequence generating $(O_t, \cdots, O_{(t+k)})$ by a recursive activation. The τ is the transition time at which C_i^d is called by C^{d-1}. Once all the recursive transitions are finished and returned to $cluster$, we get the most probable hierarchies starting from $cluster$ to the production states at T period through scanning the state list ψ, the states likelihood δ, and transition time τ.

$$L = \max_{(1 \leq r \leq N_i^d)} \left\{ \delta(\bar{t}, t+k, N_r^{d+1}, N_i^d) \, a_{End}^{N_i^d} \right\} \quad (1)$$

$$\Im = \max_{(1 \leq y \leq N^{j-1})} \left\{ \delta(t, \bar{t}-1, N_i^d, N^{d-1}) a_{End}^{N^{d-1}} L \right\} \quad (2)$$

$$stSeq = \max_{cluster} \left\{ \delta(T, cluster), \tau(T, cluster), \psi(T, cluster) \right\} \quad (3)$$

Once we have trained the model, we compare the detected hierarchies against the observed one to identify the type of workload. The hierarchies with the lowest probabilities are considered anomaly. Once we detect and identify the workload type (e.g., OL), a hierarchy of faulty states (e.g., $cluster$, N_1^2, C_1^3 and C_2^3) that is affected by the anomalous component (C_1^3) is obtained that reflects observed anomalous behaviour. We repeat these steps until the probability of the model states become fixed. Each state is correlated with time that indicates: the time of it's activation, it's activated substates, and the time at which the control returns to the calling state. The result of the fault management model (anomalous components) is

stored in Knowledge storage. This aid us in the recovery procedure as the anomalous state is recovered as first come-first heal.

C. Fault-Failure Recovery Cases

Based on the fault type, we apply a recovery mechanism that considers the dependencies between components, and the current component status. The recovery mechanism is specified based on historic and current observations of a response time for a container or node and the hidden states (containers or nodes). The following steps and concerns are considered by the recovery mechanism:

- Analysis: relies on current and historic observation.
- Observation (failure): indicates the type of observed failure (e.g., low response time).
- Anomaly (fault): reflects the fault type (e.g., overload).
- Reason: explains the causes of the problem.
- Remedial Action: explains different solutions that can be applied to solve the problem.
- Requirements: constraint that might apply.

We look at two anomaly cases and suitable recovery strategies, which exemplify recovery strategies for the fault-failure mapping cases 1.3 and 2.1. These strategies can be applied based on the observed response time (current and historic observations) and related faults (hidden states).

1) Container neighbour overload (external container dependency) **Analysis**: current/historic observations, hidden states
Observation (failure): low response time at the anomalous container and the dependent one.
Anomaly: overload in one or more containers results in underload for another container at different node.
Reason: heavily loaded container with external dependent one (communication)
Remedial Actions:

Option 1: Separate the overloaded container and the external one depending on it from their nodes. Then, create a new node containing the separated containers considering the cluster capacity. Redirect other containers that in communication to these 2 containers in the new node. Connect current nodes with the new one and calculate the probability of the whole model to know the number of transitions (to avoid the occurrence of overload) and to predict the future behaviour.

Option 2: For the anomalous container, add a new one to the node that has the anomalous container to provide fair workload distribution among containers considering the node resource limits. Or, if the node does not yet reach the resource limits available, move the overloaded container to another node with free resource limits. At the end, update the node.

Option 3: create another (MM) node within the node with anomalous container behaviour. Next, direct the communication of current containers to (MM). We need to redetermine the probability of the whole model to redistribute the load between containers. Finally, update the cluster and the nodes.

Option 4: distribute load.

Option 5: rescale node.

Option 6: do nothing, this means that the observed failure relates to regular system maintenance or update happened to the system. Thus, no recovery option is applied.
Requirements: need to consider node capacity.

2) Node overload (self-dependency) **Analysis**: current and historic observations
Observation (failure): low response time at node level.
Anomaly: overloaded node.
Reason: limited node capacity.
Remedial Actions:

Option 1: distribute load.

Option 2: rescale node.

Option 3: do nothing.
Requirements: collect information regarding containers and nodes, consider node capacity and rescale node(s).

D. Recovery Model

The recovery model (Execute stage in MAPE-K) receives an ordered list of faulty states from the identification step. It applies a recovery mechanism considering the type of the identified anomaly and the resource capacity. We have configured the fault management model to have a specific number of nodes and containers because increasing the number of nodes and containers leads to a large amount of different recovery actions (Load balancing rules), which reduces model performance.

We are mainly concerned with two workload anomalies: (1) overload as it reflects anomalous behaviour, (2) underload category, as it is considered anomaly but it represents a solution to migrate load from heavy loaded component. We define different recovery actions for each fault-failure case. Consequently, for an identified anomaly case, we need to select the most appropriate action from the time and cost perspectives.

The Recover Job Scheduler (RJS) heals the identified anomaly based on first identified-first heal. It mitigates the anomalous state, by distributing the load to the underloaded components considering their status.

The recovery actions are stored in the Knowledge storage to keep track of the number of applied actions to the identified anomalous component. Before applying any of the recovery option, "Restart" option is applied to save the cost of trying multiple recovery options if the component does not reach its restart action number limit. In case a restart option does not enhance the situation, RJS checks the existence of underloaded component identified by the fault management model and stored in the knowledge storage. If there is underloaded component, the HMM is trained using the Forward-Backward algorithm to select the most probable action for the anomalous component as shown in Figure 10. The states A_i in the model refer to the hidden recovery actions. The $rejuvenation$ hidden state refers to the restart action, and $P_i(r)$, is the probability of
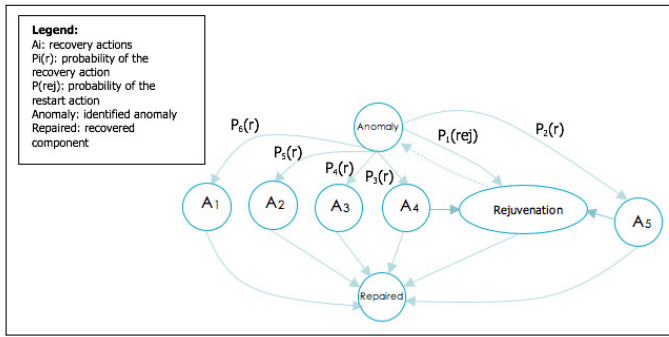
FIGURE 10. HMM FOR RECOVERY ACTIONS.

the recovery actions. We estimate $P_i(r)$ based on computing the maximum likelihood. The result of the HMM will be, for instance, the most probable action for anomalous state C_1^3 is 'distribute load'. The RJS apply the selected action to the fault component in the "Managed Component Pool". In case, RJS could not find underloaded components, the "pause" action is applied. If the number of applied recovery actions for the anomalous component exceeds a predefined threshold, a terminate action is applied after backing up the component. For each component, we further keep a profile of the type of applied action to enhance the recovery procedure in the future.

a) Metrics for Recovery Plan Determination: In order to better capture the accuracy of the proposed fault identification, we estimate the Fault Rate to capture (1) the number of faults during system execution $\Re(FN)$, and (2) the overall length of failure occurrence $\Re(FL)$ as depicted in "(4)" and "(5)". This aids us later in reducing the fault/failure occurrence through providing the best suited recovery mechanism, for instance for frequent or long-lasting failures. The observed behaviour is analysed in terms of failure rates for each state – e.g., low response times may result from overload states or normal load states – in order to determine the number of failures observed for each state and to estimate the total failure numbers for all the states. We define $\Re$ as follows:

$$\Re(FN) = \frac{No\ of\ Detected\ Faults}{Total\ No\ of\ Faults\ of\ Resource} \quad (4)$$

$$\Re(FL) = \frac{Total\ Time\ of\ Observed\ Failures}{Total\ Time\ of\ Execution\ of\ Resource} \quad (5)$$

The Average Failure Length (AFL), as in "(6)", might also be relevant to judge the relative urgency of recovery. Other relevant metrics that impact on the decision which strategy to use, but which we do not detail here, are resilience metrics addressing recovery times.

$$AFL = \frac{\sum Time\ of\ Failure\ Occurrence}{Number\ of\ Observed\ Failures} \quad (6)$$

VIII. EVALUATION

The proposed architecture runs on Kubernetes and Docker containers. We deploy the TPC-W[4] benchmark on the containers to validate the architecture. We focus on three types of faults the CPU hog, Network packet loss/latency, and performance anomaly caused by workload congestion.

A. Environment Set-Up

To evaluate the accuracy of the proposed architecture, the experiment environment consists of three VMs. Each VM is equipped with Linux OS, 3 VCPU, 2 GB VRAM, Xen 4.11 [5], and an agent. Agents are installed on each VM to collect the monitoring data from the system (e.g., host metrics, container, performance metrics, and workloads), and send them to the Real-Time/Historical storage to be processed by the Monitor. The VMs are connected through a 100 Mbps network. For each VM, we deploy two containers, and we run into them the TPC-W benchmark.

The TPC-W benchmark is used for resource provisioning, scalability, and capacity planning for e-commerce websites. The TPC-W emulates an online bookstore that consists of 3 tiers: client application, web server, and database. Each tier is installed on VM. We do not consider the database tier in the anomaly detection and identification, as a powerful VM should be dedicated to the database. The CPU and Memory utilization are gathered from the web server, while the Response time is measured from clients end. We ran the TPC-W for 300 minutes. The number of records that we obtain from the TPC-W is 2000 records.

We further use docker *stats* command to obtain a live data stream for running containers. The SignalFX Smart Agent[6] monitoring tool is used and configured to observe the runtime performance of components and their resources. We also use the Heapster[7] to group the collected data, and store them in a time series database using the InfluxDB[8]. The gathered data from the monitoring tool, and from datasets are stored in the Real-Time/Historical Data storage to enhance the future anomaly detection and identification. The gathered dataset is classified into training and testing datasets 50% for each. The model training last 150 minutes.

To simulate real anomaly scenarios, script is written to inject different types of anomalies. The anomaly injection for each component last 5 minutes. The anomaly scenarios are: (1) CPU Hog, consumes all CPU cycles by employing infinite loops. (2) Memory Leak, exhausts the component memory. The stress[9] tool is used to create pressure on the CPU and Memory.

Furthermore, workload contention is generated to test the controller under different workloads. To generate workload,

[4]http://www.tpc.org/tpcw/
[5]https://xenproject.org/
[6]https://www.signalfx.com/
[7]https://github.com/kubernetes-retired/heapster
[8]https://www.influxdata.com/
[9]https://linux.die.net/man/1/stress

TABLE I. DETECTION EVALUATION.

Metrics	HHMM	DBN	HTM
RMSE	0.23	0.31	0.26
MAPE	0.14	0.27	0.16
CDA	96.12%	91.38%	94.64%
AD	0.94	0.84	0.91
FAR	0.27	0.46	0.31

TABLE II. ASSESSMENT OF IDENTIFICATION.

Metrics	HHMM	DBN	HTM
AI	0.94	0.84	0.94
CIA	94.73%	87.67%	93.94%
IIA	4.56%	12.33%	6.07%
FAR	0.12	0.26	0.17

the TPC-W web server is emulated using client application, which generates workload (using Remote Browser Emulator) by simulating a number of user requests that is increased iteratively. Since the workload is always described by the access behaviour, we consider the container is gradually loaded within [30-2000] emulated users requests, and the number of requests is changed periodically. To measure the number of requests and response (latency), the HTTPing[10] is installed on each node. Also, the AWS X-Ray[11] is used to trace of the request through the system.

B. The Detection Assessment

The detection model is evaluated by the Root Mean Square Error (RMSE), Mean Absolute Percentage Error (MAPE), and False Alarm Rate (FAR), which are the commonly used metrics [32] for evaluating the quality of detection. We further measure the Number of Correctly Detected Anomaly (CDA) and Accuracy of Detection (AD).

a) Root Mean Square Error (RMSE): It measures the differences between the detected value and the observed one by the model. A smaller RMSE value indicates a more effective detection scheme.

b) Mean Absolute Percentage Error (MAPE): It measures the detection accuracy of a model. Both the RMSE and MAPE are negatively-oriented scores, which means lower values are better.

c) Number of Correctly Detected Anomaly (CDA): It measures the percentage of the correctly detected anomalies to the total number of detected anomalies in a given dataset. High CDA indicates the model is correctly detected anomalous behaviour.

d) Accuracy of Detection (AD): It measures the completeness of the correctly detected anomalies to the total number of anomalies in a given dataset. Higher AD means that fewer anomaly cases are undetected.

e) False Alarm Rate (FAR): The number of the normal detected component, which has been misclassified as anomalous by the model.

The efficiency of the model is compared with the Dynamic Bayesian network (DBN), see Table I. The results show that the HHMM and HTM model detects anomalous behaviour with promised results comparing to the DBN.

C. The Identification Assessment

The accuracy of the results is compared with the Dynamic Bayesian Network (DBN), and Hierarchical Temporal Memory (HTM), and it is evaluated based on different metrics such as: the Accuracy of Identification (AI), Number of Correctly Identified Anomaly (CIA), Number of Incorrectly Identified Anomaly (IIA), and FAR.

a) Accuracy of Identification (AI): It measures the completeness of the correctly identified anomalies to the total number of anomalies in a given dataset. Higher AI means that fewer anomaly cases are un-identified.

b) Number of Correctly Identified Anomaly (CIA): It is the number of correct identified anomaly (NCIA) out of the total set of identification, which is the number of correct Identification (NCIA) + the number of incorrect Identification (NICI)). The higher value indicates the model is correctly identified anomalous component.

$$CIA = \frac{NCIA}{NCIA + NICI} \tag{7}$$

c) Number of Incorrectly Identified Anomaly (IIA): The IIA is the number of the identified component, which represents an anomaly but misidentified as normal by the model. The lower value indicates that the model correctly identified anomaly component.

$$IIA = \frac{FN}{FN + TP} \tag{8}$$

d) False Alarm Rate (FAR): The number of the normal identified component, which has been misclassified as anomalous by the model.

$$FAR = \frac{FP}{TN + FP} \tag{9}$$

The false positive (FP) means the detection/identification of anomaly is incorrect as the model detects/identifies the normal behaviour as anomaly. True negative (TN) means the model can correctly detect and identify normal behaviour as normal.

As shown in Table II, the HHMM and HTM achieved promising results for the identification of anomaly. While the results of the DBN a little bit decayed for the CIA with approximately 7% than the HHMM, and 6% than the HTM. Both the HHMM and HTM show higher identification accuracy as they are able to identify temporal anomalies in the dataset. The result interferes that the HHMM is able to link the observed failure to its hidden workload.

---

[10]https://www.vanheusden.com/httping/
[11]https://aws.amazon.com/xray/

TABLE III. RECOVERY EVALUATION.

Evaluation Metrics	Results
RA	99%
MTTR	60 seconds
OA	97%

D. The Recovery Assessment

To assess the recovery decisions of the model, we measure: (1) the Recovery Accuracy (RA) to be the number of successfully recovered anomalies to the total number of identified anomalies, (2) the Mean Time to Recovery (MTTR), the average time that the architecture takes to recover starting from the anomaly injection until recovering it. (3) The Overall Accuracy (OA) to be the number of correct recovered anomalies to the total number of anomalies. The results in Table III show that once the HMM model is configured properly, it can efficiently recover the anomalies with an accuracy of 99%.

IX. BEYOND PERFORMANCE: TRUST ANOMALIES

Traditionally, anomaly detection and analysis is addressing performance and resource management if applied to software systems management. Another wide area is security and trust. An open system has security vulnerabilities. Checking continuously for anomalies showing unusual behaviour that might indicate attacks or loss of information in some form is here also an important anomaly detection objective. Trust is here a related concern that covers security as well as performance and other technical factors.

Trust problems occur if different providers and consumers of services meet in a context without any prior trust relationship. A trust anomaly here is any situation in which the delivery of a previously guaranteed service (or its quality) is in doubt. An anomaly detection solution can help here to proactively invoke a remedial action or to record more information (in a tamper-proof way) to allow for later analysis and resolution of disputes.

So, we briefly discuss here the handling of trust regarding QoS compliance using a trust anomaly detection architecture. Since trust does not exist, it is important to capture and store relevant information in a tamper-proof way. The use of blockchains as a reaction to anomalies is here an option, if a consumer needs trustworthy documentation in failure cases, but blockchains maybe also always be used if a provider need assurance about having provided as planned/promised in contract. A blockchain is a distributed data store for digital transactions, resembling a ledger [76]. Blockchains have been used for various applications [50],[52],[53],[54],[55]. The blocks are linked and secured using cryptography. Each block typically contains a cryptographic hash of the previous block, a timestamp and transaction data. By design, a blockchain is inherently tamper-proof, i.e., resistant to modification of stored data.

In more concrete terms, an anomaly detection solution as we discuss here could, if QoS compliance is under threat, then switch on blockchains [45],[49]. This could act as remedial (support) action for later analysis and providing tamper-proof information for recovery.

We have not implemented this solution yet as part of our anomaly management solution, but we want to point out with this discussion that the solution presented is not limited to performance concerns and immediate remedial actions only.

X. CONCLUSIONS

Service management in virtualised, third-party environments has both benefits and limitations. Virtualisation allows resources assigned to application services to be adjusted dynamically to meet changing need. However, the reason for changes is often hidden from the service consumer. We present a controller architecture for the detection, and recovery of anomalies in hierarchically organised clustered computing environments. We pay specific attention to recent container cluster orchestration tools like Kubernetes or Docker Swarm that are now widely used to deploy software [30].

Our key objective here is to provide an analysis feature, that maps observable quality concerns onto hidden resources in a hierarchical clustered environment, and their operation in order to identify the reason for performance degradations and other anomalies [1],[68]. From this, a recovery strategy that removes the workload anomaly, thus removing the observed performance failure is the second step. We have proposed to use the Hidden Markov Models (HMMs) to reflect the hierarchical nature of the unobservable resources, and to support the detection, identification, and recovery of anomalous behaviours. We have further analysed mappings between observations and resource usage based on a clustered container scenario. The objective of this paper is to introduce the complete controller architecture with its key processing steps. Specifically, we try here to motivate the context of container-based deployment, illustrating different use case scenarios in more detail.

As part of our future, we will continue to complete the current controller implementation. Here, further simulations and experimental evaluations are planned [44],[75],[76].

With the focus on containers, we will also address practical concerns such as the relevance for microservice architectures as an architectural style [12]. Microservices are often linked to their container-based deployment. Here it would be worth investigating to which extend common microservice architectural patterns cloud influence the occurrence of anomalies in systems in which microservices are deployed as containers [13],[15],[64]. A stronger emphasis shall also be given to self-adaptive systems [14],[16],[66],[73].

Another direction is to link observed failures more into the context of the user. Here, adding semantics [71],[74] would help to link an observed anomaly to the processes and changing circumstances a user is involved in. We are thinking here of looking at educational technology systems [57],[58],[59],[61],[63],[60], where anomalies and failure are not just technical aspects, but might impact on cognitive processes as well.

REFERENCES

[1] A. Samir and C. Pahl, "A Controller Architecture for Anomaly Detection, Root Cause Analysis and Self-Adaptation for Cluster Architectures," in *The Eleventh International Conference on Adaptive and Self-Adaptive Systems and Applications, ADAPTIVE*, 75–83. 2019.

[2] C. Pahl, P. Jamshidi, and O. Zimmermann, "Architectural principles for cloud software," in *ACM Transactions on Internet Technology (TOIT)*, 18 (2), 17. 2018.

[3] D. von Leon, L. Miori, J. Sanin, N. El Ioini, S. Helmer, and C. Pahl, "A Lightweight Container Middleware for Edge Cloud Architectures," in *Fog and Edge Computing: Principles and Paradigms*, 145-170. 2019.

[4] C. Pahl, I. Fronza, N. El Ioini, and H. Barzegar, "A Review of Architectural Principles and Patterns for Distributed Mobile Information Systems," in *14th Intl Conf on Web Information Systems and Technologies*. 2019.

[5] X. Chen, C.-D. Lu, and K. Pattabiraman, "Failure prediction of jobs in compute clouds: A google cluster case study," in *International Symposium on Software Reliability Engineering, ISSRE*, 167–177. 2014.

[6] G. C. Durelli, M. D. Santambrogio, D. Sciuto, and A. Bonarini, "On the design of autonomic techniques for runtime resource management in heterogeneous systems," Doctoral dissertation, Politecnico di Milano. 2016.

[7] T. Wang, J. Xu, W. Zhang, Z. Gu, and H. Zhong, "Self-adaptive cloud monitoring with online anomaly detection," in *Future Gen Comp Syst*, 80, 89-101. 2018.

[8] A. Samir and C. Pahl, "Anomaly Detection and Analysis for Clustered Cloud Computing Reliability," in *The Tenth International Conference on Cloud Computing, GRIDs, and Virtualization*, 110–119. 2019.

[9] P. Jamshidi, A. Sharifloo, C. Pahl, H. Arabnejad, A. Metzger, and G. Estrada, "Fuzzy self-learning controllers for elasticity management in dynamic cloud architectures," in *12th Intl ACM SIGSOFT Conference on Quality of Software Architectures, QoSA*, 70–79. 2016.

[10] P. Jamshidi, A. Sharifloo, C. Pahl, A. Metzger, and G. Estrada, "Self-learning cloud controllers: Fuzzy q-learning for knowledge evolution," in *The International Conference on Cloud and Autonomic Computing*, September. 208-211. 2015.

[11] S. Fine, Y. Singer, and N. Tishby, "The hierarchical hidden markov model: analysis and applications," in *Machine Learning*, vol. 32, no. 1, 41–62, 1998.

[12] D. Taibi, V. Lenarduzzi, and C. Pahl, "Microservices Anti-Patterns: A Taxonomy," in *Microservices – Science and Engineering*, Springer. 2019.

[13] D. Taibi, V. Lenarduzzi, and C. Pahl, "Architecture Patterns for a Microservice Architectural Style," in *Communications in Comp and Inf Science*, Springer. 2019.

[14] H. Arabnejad, C. Pahl, G. Estrada, A. Samir, and F. Fowley, "A fuzzy load balancer for adaptive fault tolerance management in cloud platforms," in *European Conference on Service-Oriented and Cloud Computing*,

[15] D. Taibi, V. Lenarduzzi, and C. Pahl, "Architectural Patterns for Microservices: A Systematic Mapping Study," in *Proceedings CLOSER Conference*, 221–232. 2018.

[16] H. Arabnejad, C. Pahl, P. Jamshidi, G. Estrada, "A comparison of reinforcement learning techniques for fuzzy cloud auto-scaling," in *17th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing*. 2017.

[17] T. F. Düllmann, "Performance anomaly detection in microservice architectures under continuous change," Master, University of Stuttgart, 2016.

[18] C. Pahl, N. El Ioini, and S. Helmer, "A Decision Framework for Blockchain Platforms for IoT and Edge Computing," in *3rd International Conference on Internet of Things, Big Data and Security* , 105-113. 2018.

[19] A. Samir and C. Pahl, "Detecting and Predicting Anomalies for Edge Cluster Environments using Hidden Markov Models," in *The Fourth IEEE International Conference on Fog and Mobile Edge Computing*, 21–28. 2019.

[20] D. Taibi, V. Lenarduzzi, and C. Pahl, "Processes, motivations, and issues for migrating to microservices architectures: An empirical investigation," in *IEEE Cloud Computing*, 4 (5), 22-32. 2017.

[21] M. Peiris, J. H. Hill, J. Thelin, S. Bykov, G. Kliot, and C. Konig, "PAD: Performance anomaly detection in multi-server distributed systems," in *International Conf on Cloud Computing, CLOUD*, June, 769–776. 2014.

[22] N. Sorkunlu, V. Chandola, and A. Patra, "Tracking system behaviour from resource usage data," in *International Conference on Cluster Computing*, 410–418. 2017.

[23] C. Pahl, "An ontology for software component matching," in *International Conference on Fundamental Approaches to Software Engineering*, 6–21. 2003.

[24] S. Maurya and K. Ahmad, "Load Balancing in Distributed System using Genetic Algorithm," in *Intl Journal of Engineering and Technology*, 5(2), 139–142. 2013.

[25] H. Sukhwani, "A survey of anomaly detection techniques and hidden markov model," in *International Journal of Computer Applications*, vol. 93, no. 18, 26–31. 2014.

[26] R. Heinrich, A. van Hoorn, H. Knoche, F. Li, L. E. Lwakatare, C. Pahl, S. Schulte, and J. Wettinger, "Performance engineering for microservices: research challenges and directions," in *ACM/SPEC International Conference on Performance Engineering Companion*, 223–226. 2017.

[27] N. Ge, S. Nakajima, and M. Pantel, "Online diagnosis of accidental faults for real-time embedded systems using a hidden Markov model," in *Simulation*, 91(19):851-868. 2016.

[28] C. Pahl, A. Brogi, J. Soldani, and P. Jamshidi, "Cloud container technologies: a state-of-the-art review," in *IEEE Transactions on Cloud Computing*. 2018.

[29] G. Brogi, "Real-time detection of advanced persistent threats using information flow tracking and hidden markov," Doctoral dissertation. 2018.

[30] D. von Leon, L. Miori, J. Sanin, N. El Ioini, S. Helmer,

September, 109-124. 2017.

and C. Pahl, "A performance exploration of architectural options for a middleware for decentralised lightweight edge cloud architectures," in *CLOSER Conference*. 2018.

[31] IEEE, "IEEE standard classification for software anomalies (IEEE 1044 - 2009)," 1–4. 2009.

[32] K. Markham, "Simple guide to confusion matrix terminology," 2014.

[33] B. Magableh and M. Almiani, "A Self Healing Microservices Architecture: A Case Study in Docker Swarm Cluster," in *International Conference on Advanced Information Networking and Applications*, 846–858. 2019.

[34] A. Khiat, "Cloud-RAIR: A Cloud Redundant Array of Independent Resources," in *The Tenth International Conference on Cloud Computing, GRIDs, and Virtualization CLOUD COMPUTING*, May, 133–137. 2019.

[35] M. Hasan, M. Milon Islam, I. Islam, and M. Hashem, "Attack and Anomaly Detection in IoT Sensors in IoT Sites Using Machine Learning Approaches," in *Internet of Things*, vol. 7, 1–14. 2019.

[36] A. Jindal, V. Podolskiy, and M. Gerndt, "Performance modelling for Cloud Microservice Applications," in *Intl Conf on Performance Engineering*, 25–32. 2019.

[37] P. Jamshidi, C. Pahl, and N. C. Mendonca, "Pattern-based multi-cloud architecture migration," *Software: Practice and Experience*, 47 (9), 1159-1184. 2017.

[38] P. Jamshidi, C. Pahl, N. C. Mendonca, J. Lewis, and S. Tilkov, "Microservices: The Journey So Far and Challenges Ahead," in *IEEE Software*, 35 (3), 24-35. 2018.

[39] C. Sauvanaud, M. Kaâniche, K. Kanoun, K. Lazri, and G. Da Silva Silvestre, "Anomaly detection and diagnosis for cloud services: Practical experiments and lessons learned," in *Journal of Syst and Softw*, 139, 84-106. 2018.

[40] P. Jamshidi, C. Pahl, and N. C. Mendonca, "Managing uncertainty in autonomic cloud elasticity controllers," in *IEEE Cloud Computing*, 50–60. 2016.

[41] C. Pahl and B. Lee, "Containers and clusters for edge cloud architectures - A technology review," in *IEEE International Conference on Future Internet of Things and Cloud*, 379–386. 2015.

[42] C. Pahl., N. El Ioini, S. Helmer, and B. Lee, "An architecture pattern for trusted orchestration in IoT edge clouds," in *The Third International Conference on Fog and Mobile Edge Computing, FMEC*, April, 63–70. 2018.

[43] N. Kratzke, "About Microservices, Containers and their Underestimated Impact on Network Performance," in *CoRR*, vol. abs/1710.0. 2017.

[44] F. Ghirardini, A. Samir, I. Fronza, and C. Pahl, "Performance Engineering for Cloud Cluster Architectures using Model-Driven Simulation," in *ESOCC Workshops - CloudWays'2018*. 2019.

[45] C. Pahl, N. El Ioini, S. Helmer, and B. Lee, "A Semantic Pattern for Trusted Orchestration in IoT Edge Clouds," in *Internet Technology Letters*. 2019.

[46] R. Scolati, I. Fronza, N. El Ioini, A. Samir, and C. Pahl, "A Containerized Big Data Streaming Architecture for Edge Cloud Computing on Clustered Single-Board

Devices," in *10th International Conference on Cloud Computing and Services Science*, 68-80. 2019.

[47] A. Wert, "Performance problem diagnostics by systematic experimentation," PhD, KIT. 2015.

[48] Q. Guan, C. C. Chiu, and S. Fu, "CDA: A cloud dependability analysis framework for characterizing system dependability in cloud computing infrastructures," in *Proceedings of IEEE Pacific Rim International Symposium on Dependable Computing, PRDC*, vol. 18, 11–20. 2012.

[49] N. El Ioini and C. Pahl, "Trustworthy Orchestration of Container Based Edge Computing Using Permissioned Blockchain," in *The Fifth International Conference on Internet of Things: Systems, Management and Security, IoTSMS*, October, 147-154, IEEE Press. 2018.

[50] G. D'Atri, V. T. Le, C. Pahl, and N. El Ioini, "Towards Trustworthy Financial Reports Using Blockchain," in *Proceedings Tenth International Conference on Cloud Computing, GRIDs, and Virtualization*. 2019.

[51] N. El Ioini and C. Pahl, "A Review of Distributed Ledger Technologies," in *On the Move to Meaningful Internet Systems. OTM 2018 Conferences*, 227-288. 2018.

[52] C. A. Ardagna, R. Asal, E. Damiani, N. El Ioini, and C. Pahl, "Trustworthy IoT: An Evidence Collection Approach Based on Smart Contracts," in *2019 IEEE International Conference on Services Computing (SCC)*, 46–50. 2019.

[53] V. T. Le, C. Pahl, and N. El Ioini, "Blockchain Based Service Continuity in Mobile Edge Computing," in *6th International Conference on Internet of Things: Systems, Management and Security*, 2019.

[54] C. A. Ardagna, R. Asal, E. Damiani, T. Dimitrakos, N. El Ioini, and C. Pahl, "Certification-based cloud adaptation," in *IEEE Transactions on Services Computing*. 2018.

[55] S. Helmer, M. Roggia, N. El Ioini, and C. Pahl, "EthernityDB - Integrating Database Functionality into a Blockchain," in *European Conference on Advances in Databases and Information Systems*, 37–44. 2019.

[56] S. Helmer, C. Pahl, J. Sanin, L. Miori, S. Brocanelli, F. Cardano, D. Gadler, D. Morandini, A. Piccoli, S. Salam, A. M. Sharear, A. Ventura, P. Abrahamsson, and D. T. Oyetoyan, "Bringing the cloud to rural and remote areas via cloudlets," in *Proceedings of the 7th Annual Symposium on Computing for Development*, 14. 2016.

[57] C. Pahl, "Layered ontological modelling for web service-oriented model-driven architecture," in *Europ Conf on Model Driven Architecture – Foundations and Applications*. 2005.

[58] S. Murray, J. Ryan, and C. Pahl, "A tool-mediated cognitive apprenticeship approach for a computer engineering course," in *Proceedings 3rd IEEE International Conference on Advanced Technologies*, 2-6. 2003.

[59] C. Pahl, R. Barrett, and C. Kenny, "Supporting active database learning and training through interactive multimedia," in *ACM SIGCSE Bulletin* 36 (3), 27-31. 2004.

[60] C. Kenny and C. Pahl, "Automated tutoring for a database

skills training environment," in *ACM SIGCSE Symposium 2005*, 58-64. 2003.

[61] X. Lei, C. Pahl, and D. Donnellan, "An evaluation technique for content interaction in web-based teaching and learning environments," in *3rd IEEE International Conference on Advanced Technologies*, 294-295. 2003.

[62] S. Govindan, J. Liu, A. Kansal, and A. Sivasubramaniam, "Cuanta: Quantifying Effects of Shared On-chip Resource Interference for Consolidated Virtual Machines," in *ACM Symposium on Cloud Computing*, 1–14. 2011.

[63] M. Melia and C. Pahl, "Constraint-based validation of adaptive e-learning courseware," in *IEEE Transactions on Learning Technologies* 2(1), 37-49. 2009.

[64] D. Taibi, V. Lenarduzzi, C. Pahl, and A. Janes, "Microservices in agile software development: a workshop-based study into issues, advantages, and disadvantages," in *XP2017 Scientific Workshops*, 2017.

[65] L. Mariani, C. Monni, M. Pezze, O. Riganelli, and R. Xin, "Localizing Faults in Cloud Systems," in *11th International Conference on Software Testing, Verification and Validation*, 262–273. 2018.

[66] N. C. Mendonca, P. Jamshidi, D. Garlan, and C. Pahl, "Developing Self-Adaptive Microservice Systems: Challenges and Directions," in *IEEE Software*. 2020.

[67] Y. Tan, H. Nguyen, Z. Shen, and X. Gu, "PREPARE: Predictive Performance Anomaly Prevention for Virtualized Cloud Systems," in *IEEE International Conference on Distributed Computing Systems*, 285–294. 2012.

[68] A. Samir and C. Pahl, "Anomaly Detection and Analysis for Clustered Cloud Computing Reliability," in *International Conference on Cloud Computing, GRIDs, and Virtualization*. 2019.

[69] T. Zwietasch, "Online Failure Prediction for Microservice Architectures," Master Thesis, U Stuttgart. 2017.

[70] A. Samir and C. Pahl, "A Controller Architecture for Anomaly Detection, Root Cause Analysis and Self-Adaptation for Cluster Architectures," in *Intl Conf on Adaptive and Self-Adaptive Systems and Applications*. 2019.

[71] M. Javed, Y. M. Abgaz, and C. Pahl, "Ontology change management and identification of change patterns," in *Journal on Data Semantics* 2(2-3), 119-143. 2013.

[72] O. Ibidunmoye, T. Metsch, and E. Elmroth, "Real-time detection of performance anomalies for cloud services," in *IEEE/ACM 24th International Symposium on Quality of Service*. 2016.

[73] P. Jamshidi, C. Pahl, S. Chinenyeze, and X. Liu, "Cloud migration patterns: a multi-cloud service architecture perspective," in *Service-Oriented Computing ICSOC2014 Workshops*, 6-19. 2015.

[74] D. Fang, X. Liu, I. Romdhani, P. Jamshidi, and C. Pahl, "An agility-oriented and fuzziness-embedded semantic model for collaborative cloud service search, retrieval and recommendation," in *Future Generation Computer Systems*, 56, 11-26. 2016.

[75] F. Ghirardini, A. Samir, I. Fronza, and C. Pahl, "Model-Driven Simulation for Performance Engineering of Kubernetes-style Cloud Cluster Architectures," in *ES-OCC 2018 Workshops, PhD Symposium, EU-Projects*, 2019.

[76] N. El Ioini, C. Pahl, and S. Helmer, "A decision framework for blockchain platforms for IoT and edge computing," in *Proceedings of the 3rd International Conference on Internet of Things, Big Data and Security*. 2018.

[77] J. Ehlers, A. van Hoorn, J. Waller, and W. Hasselbring, "Self-adaptive software system monitoring for performance anomaly localization," in *IEEE International Conference on Autonomic Computing*, 197–200. 2011.

[78] R. Nathuji, A. Kansal, and A. Ghaffarkhah, "Q-clouds: Managing performance interference effects for QoS-aware clouds," in *5th European Conference on Computer Systems*, 237–250, 2010.

An Integrated Model for Content Management, Topic-oriented User Segmentation, and Behavioral Targeting

Hans-Werner Sehring

Namics – A Merkle Company
Hamburg, Germany
e-mail: hans-werner.sehring@namics.com

Abstract—**The World Wide Web is the basis for increasingly many information and interaction services. Personalization provides users with information and services that are adequately tailored to their current needs. Targeting, a form of implicit personalization of content and content presentations for groups of users, comes to broader practical use for a growing number of commercial websites. The wider adoption results from the availability of platforms that incorporate targeting. Solutions are usually built on top of content management systems or eCommerce solutions that are used for the production of websites. The rule sets that typically are required for targeting are related to content, but they are superimposed in the sense that they are not an integral part of the content model or the content itself. This paper presents an initial model that is used to study the integration of content, content visualizations, user classification, and content targeting. Potential benefits from an integrated model are manifold. In the presented approach, personalization is applied by putting content in context rather than through superimposed targeting rules. By expressing personalization rules in the same context-dependent and evolvable way as content, they can also evolve over time and can be adapted to different user contexts. On top of that, they can be defined and maintained by content editors and other users of a content management system. The models used to study this form of integrated targeting do not rely on a certain technology or implementation. The features of the Minimalistic Meta Modeling Language (M3L) are employed as a meta model testbed, though. It allows expressing personalization rules along with both the content they refer to and the concepts they are based on, as well as users' interest, in the required consistent way.**

Keywords-personalization; targeting; segmentation; web tracking; profiling; content management.

I.   INTRODUCTION

The World Wide Web has undergone a tremendous development. For over two decades now, there is research on *personalization* of contents that are published on the web and of the presentations used for its publication.

For quite some time now, personalized content publication has arrived in the practical operation of websites. *Targeting*, the form of personalization that is found in practice most often, has been studied in [1]. In this article, we extend this work by modeling and analyzing the various deduction steps that are involved in targeting in more depth.

There is a wide range of personalization approaches for different purposes and goals [2]. These approaches differ in several aspects [3], e.g., in the way personalizations are derived: explicitly by users stating their preferences or implicitly by deriving them from users' behavior and habit. An example for explicit personalization are websites that allow users to name their interests or that allow to individually rearrange parts of the web site. Implicit personalization is achieved, e.g., by observing interactions of a user with a website [4] or by taking previously visited websites into consideration (customer journeys, at best).

Targeting that is based on user behavior is centered around the assumption that users' interest can be derived from their search or browsing behavior [5].

Personalization approaches also differ in the subject of the individual adaptations, e.g., content or content representations (visualizations of content created for publication). Content personalization can be found, e.g., in online shops where users receive individual pricing. Content visualizations are personalized by, e.g., ordering lists of content entries in a user-specific way.

Personalization has already been adopted to a range of specific, innovative websites, in particular those that confront the user with large amounts of content [6][7]. Such websites use personalization to filter and prioritize content based on assumed user preferences.

Currently, *targeting* is applied by an increasing number of commercial websites. We consider targeting as implicit personalization of content for user groups. The adaptation of content is limited to predefined points, though. Typically, part of the content is selected from building blocks that are prepared for the different user groups.

A set of tools that has emerged during the past years constitutes the basis that allows configuring websites for personalization. Examples are personalization engines built into content management systems and commerce platforms, as well as external personalization services that allow adjusting websites to specific user groups.

There is a lack of models that would cover multiple kinds of personalization approaches [8] and, therefore, allow different usage scenarios to be integrated in one solution.

Typically, commercial products use means of personalization that are superimposed to a (non-individualized) base system. A content management system, e.g., allows defining a content model according to which content will be edited, managed, and published. This content

model is defined in a uniform way for all users and application scenarios. On a different layer, personalization is added by other means, typically rules that define how to adjust content representations of specific user groups.

Therefore, there is no coherence between content models, content visualization layouts, and personalization rules in such systems. Instead, content has to be defined with all possible audiences and usage scenarios in mind, visualizations have to provide the variations to be offered as personalizations, and personalization rules may only be defined within the limits set by these definitions.

Contemporary products typically require fixed content models and visualizations (or at least ones that cannot be changed by content editors). This only leaves such personalization rules at the content editors' disposal that can be defined with respect to the possibilities and constraints raised by content models and content visualizations.

The aim of this paper is providing first studies towards a fully integrated model that combines many aspects of content and its personalized utilization. For this study, we use the Minimalistic Meta Modeling Language (M3L) [9] as a testbed. This language is well-suited for content models since it covers variations and contexts of content in a direct way. Insights into a variety of personalization options originate from previous research on Concept-oriented Content Management [10]. These insights are transferred to M3L models.

The rest of this paper is organized as follows. Section II describes targeting approaches typically found in commercial software products. Section III provides a short introduction into the M3L. Section IV gives a brief outlook on an architecture of M3L implementations. Sections V to XI present the modeling experiment of utilizing the M3L for expressing and integrating the common targeting approach into content models for websites. Conclusions and acknowledgement close the paper.

II. Targeting in Commercial Products

There is a wide range of approaches to personalization that can be found in the literature and in prototype implementations. In this paper, we constrain ourselves to targeting, which is of particular importance for commercial websites. Targeting is a form of implicit personalization of content assembled for presentation with respect to a customer group. The personalization itself is directed by rules set up by content editors.

Another application of targeting is advertising, also used for so-called retargeting. In contrast to website personalization, targeted advertising allows to quantitatively measure success. In many cases, targeting turned out to significantly increase advertising success [11].

A. Segment-based Targeting Rules

For targeting, as it is found in many commercial products, users of a web site are assigned *segments*. Segments are categories describing a user's interest or preferences. These segments are predefined for a particular website [12] (though there are scientific approaches that include deriving segments by, e.g., means of clustering [13]). Segments connect website topics with user preferences.

The assignment of segments to users is based on *tracking* (or *analytics*) used during web page delivery. By tracking, accesses to web pages are recorded. Tracking can be integrated into the system that delivers the web pages, or it can be performed by an external service [14]. Depending on the granularity required, interactions on smaller parts than whole pages may be counted [15].

From the web pages visited by a user, her or his interests are derived by collecting the topics covered by those web pages. The web pages considered in this collection could be, e.g., those web pages that have been visited most often, or the web pages for which the visits exceed a given threshold.

The segments assigned to a user (by that time) are used as a parameter to content selection and to the production of documents from content. This way, content and its representations are personalized for user groups, namely groups consisting of users with the same segments assigned.

B. Related Work

Targeting is found in diverse systems and services, e.g., in Content Management Systems (CMSs), commerce systems, and marketing suites.

All commercial approaches are limited by the fact that certain parts of the targeting process are built into the solution. The means for producing targeted content are given by the software and typically consist of rules that select content based on certain parameters. Segments often cannot be changed dynamically, but are predefined during system configuration. The approach of this article aims to overcome these restrictions and to make all parts of the targeting process accessible to editors.

1) Personalization Engines in Content Management Systems. Some CMSs have means of segmentation built in. These systems allow equipping content with rules for the selection of content to be included in published web pages based on user segmentation. These rules are applied during document production. Like many others, the CMS products of CoreMedia [16] and Sitecore [17] work this way.

2) Superimposed Personalization. Instead of an integrated personalization engine inside a CMS, an external service can alternatively perform personalization. Adobe Target [18] is a prominent representative of this personalization approach. External tracking is based on published documents, not the content itself. Likewise, targeting is performed by modifying documents.

3) Consideration of Additional Information on Users. Instead of just considering user behavior in the form of web page access profiles, increasingly many applications are also based on explicit customer data. Such data come from, e.g., a Customer Relationship Management (CRM) system, from the history of transactions in a commerce system, from the history of cases in a support system, or from feedback given by means of ratings. Personalization may additionally be based on context information, e.g., the time of day, the device the visitor uses, or some kind of work mode she or

he is in [19]. Such context information is partially considered in commercial personalization engines.

III.   THE MINIMALISTIC META MODELING LANGUAGE

The *Minimalistic Meta Modeling Language* (*M3L*, pronounced "mel") is a modeling language that is applicable to a range of modeling tasks. It proved particularly useful for context-aware content modeling [20].

For the purpose of this paper, we only introduce the static aspects of the M3L in this section. Dynamic evaluations that are defined by means of different rules are presented in the subsequent section.

The descriptive power of M3L lies in the fact that the formal semantics is rather abstract. There is no particular (domain) semantics of M3L concept definitions. There is also no formal distinction between typical conceptual relationships (specialization, instantiation, entity-attribute, aggregation, materialization, contextualization, etc.).

A.   Concept Definitions and References

A M3L definition consists of a series of definitions or references. Each definition starts with a previously unused identifier that is introduced by the definition and may end with a period, e.g.:

Person.

A reference has the same syntax, but it names an identifier that has already been introduced.

We call the entity named by such an identifier a *concept*.

The keyword `is` introduces an optional reference to a *base concept*, making the newly defined concept a *refinement* of it.

A specialization relationship as known from object-oriented modeling is established between the base concept and the newly defined derived concept. This relationship leads to the concepts defined in the context (see below) of the base concept to be visible in the derived concept.

The keyword `is` always has to be followed by either `a`, `an`, or `the`. The keywords `a` and `an` are synonyms for indicating that a classification allows multiple sub-concepts of the base concept:

Peter is a **Person. John** is a **Person.**

There may be more than one base concept. Base concepts can be enumerated in a comma-separated list:

PeterTheEmployee is a **Person,** an **Employee.**

The keyword `the` indicates a closed refinement: there may be only one refinement of the base concept (the currently defined one), e.g.:

Peter is the **FatherOfJohn.**

Any further refinement of the base concept(s) leads to the redefinition ("unbinding") of the existing refinements.

Statements about already existing concepts lead to their redefinition. For example, the following expressions define the concept Peter in a way equivalent to the above variant:

Peter is a **Person.**
Peter is an **Employee.**

```
Person {
  Name is a String.
}
Peter is a Person{
  "Peter Smith" is the Name.
}
Employee {
  Salary is a Number.
}
Programmer is an Employee.
PeterTheEmployee is a Peter, a Programmer{
  30000 is the Salary.
}
```

Figure 1.   Simple M3L definitions.

B.   Content and Context Definitions

Concept definitions as introduced in the preceding section are valid in a context. Definitions like the ones seen so far add concepts to the top of a tree of contexts. Curly brackets open a new context. Figure 1 shows a sample M3L model.

We call the outer concepts the *context* of the inner, and we call the set of inner concepts the *content* of the outer.

In this example, we assume that concepts **String** and **Number** are already defined. The sub-concepts created in context are unique specializations in that context only.

As indicated above, concepts from the context of a concept are inherited by refinements. For example, Peter inherits the concept **Name** from **Person**.

M3L has visibility rules that correlate to both contexts and refinements. Each context defines a scope in which defined identifiers are valid. Concepts from outer contexts are visible in inner scopes. For example, in the above example the concept **String** is visible in **Person** because it is defined in the topmost scope. **Salary** is visible in **PeterTheEmployee** because it is defined in **Employee** and the context is inherited. **Salary** is not valid in the topmost context and in **Peter**.

Concepts that are not visible in one scope because they are only defined in a neighboring scope can be accessed with the `from` clause. For example, the **Name** of **Peter** is accessed by

Name from **Peter.**

C.   Contextual Amendments

Concepts can be redefined in contexts. This happens by definitions as those shown above. For example, in the context of **Peter**, the concept **Name** receives a new refinement.

Different aspects of concepts can explicitly be redefined in a context, e.g.:

```
AlternateWorld {
  Peter is a Musician {
    "Peter Miller" is the Name.
  }
}
```

We call a redefinition performed in a context different from that of the original definition a *conceptual amendment*.

```
Person {
  Sex.
  Status.
}
MarriedFemalePerson is a Person {
  Female is the Sex.
  Married is the Status.
}
MarriedMalePerson is a Person {
  Male is the Sex.
  Married is the Status.
}
Mary is a Person {
  Female is the Sex.
  Married is the Status.
}
```

Figure 2.   Concept refinements and a potential narrowing.

In the above example, the contextual variant of Peter in the context of **AlternateWorld** is both a **Person** (initial definition) and a **Musician** (additionally defined). The **Name** of the contextual **Peter** has a different refinement that replaces the original one.

A redefinition is valid in the context it is defined in, in sub-contexts, and in the context of refinements of the context (since the redefinition is inherited as part of the content).

D. Concept Narrowing

There are three important relationships between concepts in M3L.

M3L concept definitions are passed along two axes: through visibility along the nested contexts, and through inheritance along the refinement relationships.

A third form of concept relationship, called *narrowing*, is established by dynamic analysis rather than by static definitions like content and refinement.

For a concept c_1 to be a narrowing of a concept c_2, c_1 and c_2 need to have a common ancestor, and they have to have equal content. Equality in this case means that for each content concept of c_2 there needs to be a concept in c_1's content that has an equal name and the same base classes.

For an example, consider the definitions **MarriedFemalePerson** and **MarriedMalePerson** shown in Figure 2.

With these definitions, the concept **Mary** is a narrowing of **MarriedFemalePerson**, even though it is not a refinement of that concept, and though it introduces separate nested concepts **Female** and **Married**:

- They have the common ancestor **Person**.
- For each concept in the content of **MarriedFemalePerson** (**Female**, **Married**) there is a concept in **Mary** with the same name and common base classes

Mary is not a narrowing of **MarriedMalePerson** since it lacks a concept named **Male** in its content.

E. Semantic Rule Definitions

For each concept, one *semantic rule* may be defined.

```
MarriedFemalePerson is a Person {
  Female is the Sex.
  Married is the Status.
} |= Wife.
MarriedMalePerson is a Person {
  Male is the Sex.
  Married is the Status.
} |= Husband.
```

Figure 3.   Concept redefinitions.

The syntax for semantic rule definitions is a double turnstile ("|=") followed by a concept definition. A semantic rule follows the content part of a concept definition, if such exists.

A concept definition within a semantic rule is not made effective directly, but is used as a prototype for a concept to be created later.

The concepts shown in Figure 3 redefine the concepts **MarriedFemalePerson** and **MarriedMalePerson**.

The concepts **Wife** and **Husband** are not added to the model directly, but at the time when the parent concept is evaluated. Evaluation is covered by the subsequent section.

Concepts from semantic rules are created and evaluated in different contexts. The concept is instantiated in the same context in which the concept carrying the rule is defined. The context for the evaluation of a rule (evaluation of the newly instantiated concept, that is) is that of the concept for which the rule was defined.

In the example above, the concept **Wife** is created in the root context and is then further evaluated in the context of **MarriedFemalePerson**.

Rules are passed from one concept to another by means of inheritance. They are passed to a concept from (1) concepts the concept is a narrowing of, and (2) from base classes. Inheritance happens in this order: Only if the concept is not a narrowing of a concept with a semantic rule then rules are passed from base concepts.

For example, **Mary** as defined above evaluates to **Wife**.

F. Syntactic Rule Definitions

Additionally, for each concept one *syntactic rule* may be defined.

```
" ". ".".
WordList {
  Word.
  Remainder is a WordList.
} |- Word " " Remainder.
OneWordWordList is a WordList |- Word.
Sentence { WordList. } |- WordList ".".
HelloWorld is a Sentence {
  Words is the WordList {
    Hello is the Word.
    OneWordWordList is the Remainder {
      World is the Word.
    }
  }
}
```

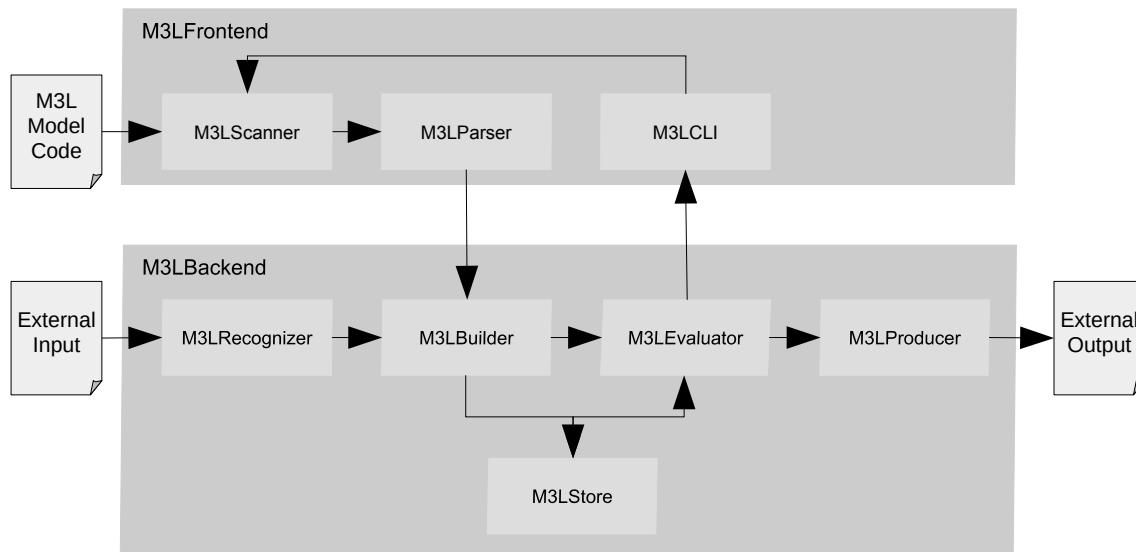Figure 4.   Syntactic rules of M3L concepts.

Figure 5. High-level architecture of the M3L system framework.

Such a rule, like a grammar definition, can be used in two ways: to produce a textual representation from a concept, or to recognize a concept from a textual representation.

A semantic rule consists of a sequence of concept references or **the name** expressions that evaluate to the current concept's name.

During evaluation of a syntactic rule, rules of referenced concepts are applied recursively. Concepts without a defined syntactic rule are evaluated to/recognized from their name.

As an example, from the definition of the concept **HelloWorld** in Figure 4, the textual representation `Hello World.` is produced. In this example, first the two concepts named blank (" ") and dot (".") are defined. These are simply used as string literals.

The concept **WordList** represents a simple linked list with entries containing one word and a reference to the next entry. The concept is defined with a syntactical rule that consists of a sequence of the concepts **Word** (the word in sequence), " ", and **Remainder** (the next entry).

The concept **OneWordWordList** represents lists with just one entry, and it overrides the syntactical rule with the respective definition. A sentence is a **WordList**, syntactically followed by a full stop (".").

HelloWorld is a concrete **Sentence**. When its syntactic form is produced, the rule inherited from **Sentence** is used. This one delegates text production to the rule of **WordList** and adds a full stop. **Words** in the content of **HelloWorld** inherits this rule and prints out the words (**Hello** and **World**) in sequence, separated by the blank. The concept for the blank has no syntactic rule, so it prints out its own name, thus producing the whitespace.

IV. AN ARCHITECTURE FOR M3L IMPLEMENTATIONS

A system for the management and evaluation of M3L models can be implemented in a straightforward way. Current work towards a M3L system implementation aims at a framework within which different system configurations can be set up.

The high-level architecture of that framework is sketched in Figure 5. The rectangle boxes represent classes of components that are filled with matching implementations in an actual system configuration. The arrows denote the flow of model representations.

The basic architecture follows that of a classical compiler. It can be separated into a frontend and a backend part. The frontend part is mainly comprised of a scanner and a parser, which read in M3L code and produce an intermediate representation of the M3L models from that code.

The frontend can optionally host a command line interface (CLI). Concrete interfaces allow evaluating lists of concept references (thus triggering their semantic rules), or they provide interactive operation by directly applying M3L statements, which a user typed in.

A *M3LBuilder* component constitutes the interface to the backend. A builder receives the intermediate representation of M3L statements and builds up the model structures in the *M3LStore*, a component that abstracts from model storage.

Another central component of the backend is a *M3LEvaluator*. This component resolves concept references in contexts and it evaluates semantic rules of the concepts. It operates on the *M3LStore*.

A *M3LStore* manages concepts that have explicitly been defined (created by a *M3LBuilder*) and ones that have been produced from rules (by a *M3LEvaluator*). A simple implementation of a store for the management of models operates in main memory. Others persist concepts in an external database system [20].

The *M3LBuilder* and in particular the *M3LEvaluator* query the store a lot when resolving concepts in their contexts. Because of the different query capabilities of different storage technologies, the framework allows optimized implementations of the builder and the evaluator that utilize the underlying store best. For example, there are different strategies to resolve concepts in the two tree structures of concept refinement and context containment.

```
WebPage.
String. Integer. FormattedText.
ContentReference.
Teaser.
OverviewPage is a WebPage {
  Title is a String.
  MainContent is a String.
  NewsTeaser is a Teaser.
}
```

Figure 6.   Base model for targetable websites.

There are two components dedicated to syntactical rules: recognizers and producers. These interface with the environment.

Producers are typically generated from syntactic rules by transforming them into a grammar definition. A grammar definition is then processed by a parser generator. The generation of grammars from syntactic rules became feasible with the advent of powerful tools such as AntLR [21].

For recognizers, templates for the production of text are generated. Template generation is feasible since concept evaluation is done by the evaluator. The templates only contain code to substitute concept names with the output of the evaluation of the referenced concepts (compare [22]). That output is in turn produced by templates.

There are multiple possible generators for recognizers and producers that operate on different input and output channels. This allows specific implementations for applications.

For example, for web-based systems, the recognizer may interface with a web server to analyze HTTP requests that are directed at concepts. A request to *http://server/A/B/C* resolves `C from B from A` in the M3L. The producer can be a templating engine that creates external output to be transmitted as the response to the HTTP request.

V.   A MODEL OF CONTENT PERSONALIZATION

The M3L allows defining structured content models, as well as content itself. Furthermore, it can be used to define layouts, for example, for web pages that define how content is rendered into documents.

In this article, we demonstrate how to add targeting to such models. The aim is to allow integrated models for content, layout, segmentation, and personalization.

Such an integrated model allows content editors to define segmentation as well as targeting rules together with content. This way, they are not restricted to a set of predefined rules for content selection for rendering, but have full modeling power. All models are put under control of the editors. They can, for example, synthesize content, change the layout, even add new renderings, etc.

With segmentation and targeting integrated in content models, all definitions are made in a coherent and consistent way. Since the difference between a content model, content, and content personalization in the M3L is merely a matter of contextual interpretation, there is much modeling liberty in setting up targeted content (presentations).

The subsequent sections incrementally introduce one way of setting up such structures. provide a first simple M3L

model of content, its visualization on web pages, website users, web page accesses, and the targeting of the web pages to the users based on past web page accesses. The sections follow the principles of targeting as follows:

- Section VI covers basic website content management and web page rendering.
- A classification of users through segments is introduced in Section VII.
- Section VIII extends the basic content management by introducing targetable websites.
- Classical web tracking is modeled in M3L in Section IX.
- Means to turn track records into a segmentation of users, here based on users' interest, are presented in Section X.
- Finally, Section XI discusses the utilization of segments to apply targeting rules on targetable websites.

Every section gives base definitions required to add a modeling feature to content management, as well as sample applications by which we can discuss these features.

VI.   A WEB PAGE MODEL

This section demonstrates simple content, content presentation, and content delivery models for a basic website. For the integrated models developed in the course of this article, we formulate these models in the M3L. On that basis, the subsequent sections add targetable website definitions, tracking, segmentation, and finally targeting.

A.   Base Definitions for Website Structure and Content

The M3L allows modeling content in itself. Therefore, no specific base model is required for content alone. For the models to be integrated, some structures are defined, though.

The sample model in Figure 6 starts with the concept `WebPage` that constitutes the root of the content model. The concept itself defines no further structure. Is serves as a base concept for refinements that contain content and that syntactically evaluate to HTML code.

From a pure content management perspective, a web page is not a proper piece of content. Content should be independent of a publication channel, layout, etc. For the sake of simplicity, though, we take this simple model as a shortcut to our examples.

In many contemporary content management systems, base types for parts of content are predefined. Content types are defined as Cartesian products from these base types. Example for such base types are: string, formatted text, integer, and content reference, indicated in Figure 6.

Web pages are filled with components. Again, a (graphical) component is not the right abstraction for pure content management since it refers to a certain layout, but we take this shortcut here. As one sample component type, Figure 6 lists a `Teaser`. Such a component shows a preview of some content, e.g., an abstract of a text, and it serves as a reference to it.

A typical content management system is furthermore parameterized with page types that name the kinds of web

pages that are produced from such a system. Examples are the front page, detail pages, overview pages, search pages, etc. Figure 6 introduces an **OverviewPage** as the basis for all overview pages.

Actual web pages are defined as refinements of the **WebPage** concept or, in the case of overview pages, as refinements of the **OverviewPage** concept.

By this definition, all overview pages have a title, main content, and a news teaser. A news teaser is some teaser that displays some current information. All content parts can be refined to specific content on refinements for concrete pages. The news teaser, though, will below be used for targeting.

B. A Sample Set of Web Pages

Figure 7 shows two sample overview web pages for two parts of the web page. For sports topics, the concrete overview web page **SportsOverviewPage** is defined. Likewise, the **CulturePage** web page directs the user to the parts of the website that cover cultural topics.

C. Document Rendering

In our model, document creation (rendering) is represented by syntax rules that are attached to **WebPage**s.

If the output shall be a web page, definitions of concepts for HTML elements are needed as a basis for HTML document creation. Figure 8 shows such definitions inside the **HTML** context.

One level of concepts in the hierarchy of **WebPage** concepts can be considered a *web page template*, a blueprint for the creation of HTML documents for a range of web pages. Such a template is typically found in (website) production systems.

In our example, the above **OverviewPage** can be the template for all overview pages for the sections. Figure 8 shows a syntactical rule for **OverviewPage**.

In practical systems, typically more than one output is generated from content. In this case, not only one rendering for a web page is given. Applications are manifold: Multichannel publishing allows to output, e.g., a web page, print documents, and a mobile app from the same content. Multisite publishing allows to produce more than one website from a content base.

In order to define more than one output format for the same content in M3L, syntax rules need to be defined in contexts. Figure 8 indicates definitions of site-specific renderings in the contexts **WebSite1** and **WebSite2**.

```
SportsOverviewPage is an OverviewPage {
  Sports is the Title.
  "On this page..." is the MainContent.
}
CulturePage is an OverviewPage {
  "Museums and Exhibitions" is the Title.
  ReportOnNewExhibition is the MainContent.
}
ReportOnNewExhibition is a RichText {
  ...
}
```

Figure 7.   Sample overview web pages.

```
HTML {
  <html>. </html>.
  <head>. </head>.
  <title>. </title>.
  <body>. </body>.
  ...
}
HTML {
 OverviewPage |- <html>
  <head>
    <title> Title </title>
  </head>
  <body>
    ... Title ...
    ... MainContent ...
    ... NewsTeaser ...
  </body>
  </html>. }
WebSite1 is an HTML {
  OverviewPage |- ...
}
WebSite2 is an HTML {
  OverviewPage |- ...
}
```

Figure 8.   Example of a web page template.

VII.   USER SEGMENTS

In order to be able to define targetable web page templates, we first need to introduce segments as predefined clusters of users with a particular interest. Segments are important for understanding user behavior and for the targeting process.

A. Base Definition for Segments

As the basis for targeting, we just define one concept **Segment** that serves as a base concept for all clusters of users. It is the means to derive user interest from user behavior, and it describes the topics of web pages.

In practice, there are other factors that influence targeting. One such factor are context parameters that take, for example, a user's device into account.

Often, also historical user data is considered. Such data may be, for example, past purchases on an eCommerce site or an explicitly maintained user profile that the user has created by registering an account for the site.

Such additional factors are not discussed here.

B. Example of Segment Definitions

Segments might be managed in a structure like shown in the example in Figure 9, assuming for our example a news site consisting of sections.

```
Segments {
  Sections {
    Politics is a Segment.
    Sports is a Segment.
  }
}
```

Figure 9.   Samples segments.

Only the concrete segments **Politics** and **Sports** are significant. The contexts **Segments** and **Sections** are used to structure the set of concrete segments.

Segments are used in a twofold manner: On web page accesses, they name the topic of a web page in order to derive the area of interest of a visitor. When delivering a web page in a personalized way, a user's segment is used to select and evaluate personalization rules. These two uses will be elaborated below.

VIII. TARGETABLE WEBSITES

Since we aim at an integrated model, we take the approach of targeting that is based on personalized content selection and adaptation during document creation.

For that approach, content needs to be structured in a way that allows changing or rearranging it. On that basis, variations of content for specific user groups can be defined.

A. Base Definition for Targetable Websites

Using the M3L, personalization can be expressed in a straightforward way by providing variants of web page definitions in different contexts. Rules as used in commercial products are not required. In the case of targeting, we use the user segments as context of content variants.

This way, personalization is performed by choosing one out of different variants of a web page depending on the context of the requesting user.

As a basis for targetable web pages, we define a concept **TargetableWebPage** as a refinement of the concept **WebPage**. Like **WebPage** for conventional websites, the new base concept is not required for targeting to work. But it allows us to add some structure to our set of web page definitions. In particular, it allows distinguishing between web pages for which to apply targeting rules and for those for which not to apply them.

For all web pages, which shall be targeted, a corresponding concept is defined as a refinement. We demonstrate this using an example.

B. Targetable Website Example

To continue with the running example, we introduce a concept **TargetableOverviewPage** as the base of all overview pages in Figure 10.

As an example for a specific page, the concept **CulturePage** introduced in Figure 7 is redefined in this hierarchy. This way, it becomes subject to targeting.

The **Teaser** that we introduced as an example of a component is used for targeting in Figure 10. The example lists three such teasers to be used for users who are interested in politics or sports, respectively.

Concepts for the choice of teasers to use for specific user segments is shown below in Figure 10. Using the segments from above, these teasers can be defined as the news teaser of overview pages in the context of a specific segment.

With the definitions of this example, the news teaser referenced from overview pages evaluates to the variant for a specific segment when the page is evaluated in the context of that segment.

```
TargetableWebPage is a WebPage.

TargetableOverviewPage is a
  TargetableWebPage, an OverviewPage.
CulturePage is a TargetableOverviewPage.

LatestPollResults is a Teaser {
  ... Title ... TeaserText ... WebPage.
}
SoccerExhibition is a Teaser {
  ... Title ... TeaserText ... WebPage.
}
RunningGameScore is a Teaser {
  ... Title ... TeaserText ... WebPage.
}

Segments {
  Sections {
    Politics {
      TargetableOverviewPage {
        LatestPollResults
          is the NewsTeaser.
      }
    }
    Sports {
      TargetableOverviewPage {
        SoccerExhibition is a NewsTeaser.
        RunningGameScore is a NewsTeaser.
      }
    }
  }
}
```

Figure 10. Example of a targetable website.

Note that this holds for any overview page that allows targeting since the **NewsTeaser** refinements are made in the context of the base concept **TargetableOverviewPage**.

IX. TRACKING WEB PAGE VISITS

Web tracking is applied to websites for quite a while. Originally it was used to measure the usage of websites and it served to generate reports that allow analyzing users' behavior.

Today's tracking serves further purposes. In our case, it builds the foundation for segmentation by providing the basic usage information used to derive user behavior and, finally, user interest.

To integrate a model of tracking, we add some base definitions and add it to our running example.

A. Base Definitions for User Tracking

Figure 11 shows the base definitions **User** and **Visit** that we employ for tracking.

```
User.
Visit {
  Visitor is a User.
  ViewedPage is a WebPage.
}
```

Figure 11. Base model for tracking.

```
Anon815 is a User.
Anon815-on-SportsOverviewPage-1 is a Visit {
  Anon815 is the Visitor.
  SportsOverviewPage is the ViewedPage.
}
```

Figure 12. Example of tracking.

The **User** concept serves as the identity of a web page visitor. It may contain user data as content.

A **Visit** records the access of a user to a web page. A **Visit** thus refers to the requested page and to the user who requested it. If the user is unknown, we create a **User** concept refinement at the time of the first request. The newly created concept represents one anonymous user.

In real-world applications, typically a tracking tool is used for this purpose. In M3L implementations, **Visit** refinements are created on web page access (e.g., during HTTP request handling as discussed in Section IV as an example of a specific recognizer).

Targeting is based on the users' behavior. Behavior is analyzed by tracking web page accesses. In the example of the M3L model we do so by creating (or finding) a matching **Visit** instance for a web page and a user.

B. Example of a Web Tracking Activity

With the sample definitions made so far, assume a user is requesting the page described by **SportsOverviewPage**.

On first access of that web page, a concept **Anon815** is created as a refinement of **User** for the request. For every access of that user to that web page, a **Visit** concept for that user and the sports overview page is created as shown in Figure 12.

The suffix **-1** represents some part of the generated concept name that is used to disambiguate instances. In this example by a counter going up.

If **Anon815** next requests **CulturePage** in this example, then the system defines

```
Anon815-on-CulturePage-1 is a Visit {
  Anon815 is the Visitor.
  CulturePage is the ViewedPage.
}
```

Another request of user **Anon815** to the web page **SportsOverviewPage** would lead to the creation of an equivalent concept named **Anon815-on-SportsOverviewPage-2**. This way, every such request is tracked.

With this sample requests, the statement

```
Visit { Anon815 is the Visitor. }
```

evaluates to { **Anon815-on-SportsOverviewPage-1**, **Anon815-on-CulturePage-1**, **Anon815-on-SportsOverviewPage-2** }.

```
SegmentingWebPage is a WebPage {
  Topic is a Segment.
}
```

Figure 13. Base model for segmenting web pages.

```
SportsOverviewPage is a SegmentingWebPage {
  Sports from Sections from Segments
    is the Topic.
}
```

Figure 14. Example of a segmenting web page.

The curly brackets are not M3L syntax here, they denote a mathematical set. Since the referenced **Visit** evaluates to more than one concept, the result is a set. Further evaluation will continue element-wise.

X.   USER SEGMENTATION

Based on the track records represented by **Visit** instances, the user segmentation is derived. Users shall be assigned to clusters that reflect their particular interest.

Segments can be defined in many ways. In this article, we concentrate on the analysis of user behavior as provided by tracking with respect to the segments defined for a website.

Segments are derived from visits by analyzing the topics of the visited pages. The approach is based on the assumption that the topics of the web pages that a user visits most often are the ones that the user is interested in.

A. User Segmentation Based on Interest

In order to turn track records into segment assignments this way, we add segmentation information in order to describe the web pages' meaning. More precisely, for those web pages we consider relevant for identifying the users' interest, we add the according segment to a web page definition.

To this end, we add a new concept **SegmentingWebPage** as shown in Figure 13. This concept serves as a new base concept for those web pages that are used to identify the interest of a user.

Each **SegmentingWebPage** has a **Topic** assigned. The topic is to be refined to a concrete **Segment** as defined in Section VII.

As an example for a **SegmentingWebPage**, we redefine the sports overview page in a way that helps to determine the interests of a user in Figure 14.

This way, every user requesting the sports overview page is a potential member of the **Sports** segment.

B. Determining Segments From Track Records by Scoring

Segments are assigned to users on the basis of tracked accesses to segmenting web pages. Segment information is determined in a two-step process: First, assign a relevance *score* to every user for every segment by counting accesses of that user to web pages of that segment. Second, determine to which segment(s) a user belongs. The segments that have the highest score for a particular user are assigned to that user.

The score a segment got for a user is the number of visits of a user to web pages with a topic that equals that segment. In order to compute the number of visits recorded by tracking, we need to introduce the base concept **Integer** with just enough conception in order to have the ability to

count. To this end, **Integer**s have a reference **Pred** to their predecessor. Using this reference, the order of integers is defined. The numerical value of an **Integer** is thus the length of the chain of its predecessors. In M3L:

```
Integer { Pred is an Integer. }
0 is an Integer.
1 is an Integer { 0 is the Pred. }
```

On the basis of this **Integer** concept, we can define **Score** as shown in Figure 15. The **Value** of a **Score** that a segment has for a user is assigned an **Integer** concept as a refinement.

In the context of a user segment, the concept **Visit** used for tracking is redefined so that the visited pages are constrained to **SegmentingWebPage**s, and that every **Visit** created during web tracking directly leads to an update of a score.

The **Visit** directly creates a **Score** with the given user and segment through a rule. The score is set by the two refinements of **Score**.

Any score is a **ScoreUpdate**, so the value of a newly created score will be initialized with the specific **Integer 1**.

If such a score already exists with the given user and the web page's topic assigned (recognized by **Value** being an **Integer**), then it will be narrowed to the matching **ScoreIncrement**.

That concept in return will increment the value by evaluating to a **Score** with an **Integer** value that is the successor of the current value.

Else, the semantic rule will initialize the score by setting the **Value** to the **Integer 1**.

```
Score {
  SegmentedUser is a User.
  AssignedSegment is a Segment.
  Value.
}
Visit {
  ViewedPage is a SegmentingWebPage.
} |= Score {
  Visitor is the SegmentedUser.
  Topic from ViewedPage
    is the AssignedSegment.
}
ScoreUpdate is a Score
|= Score {
  SegmentedUser is the SegmentedUser.
  AssignedSegment is the AssignedSegment.
  1 is the Value.
}
ScoreIncrement is a ScoreUpdate {
  Value is an Integer.
}
|= Score {
  SegmentedUser is the SegmentedUser.
  AssignedSegment is the AssignedSegment.
  Integer { Value is the Pred. }
    is the Value.
}
```

Figure 15. Example of targeting definitions.

```
SegmentDetermination {
  InitialThreshold is an Integer.
  SegmentsOfUser {
    UserToSegment is a User.
  }
  |= Score {
    UserToSegment is the SegmentedUser.
  }
  Score_rec is a Score {
    Value is an Integer.
  } |= Score {
    Pred from Value is the Value.
    Pred from Threshold is the Threshold.
  }
  IncludedScore is a Score_rec {
    0 is the Threshold.
  } |= AssignedSegment.
  ExcludedScore is a Score_rec {
    0 is the Value.
  } |= Segments.
}
```

Figure 16. Base model for segmentation.

On every request of a user u for a web page p, the web server issues a

```
Visit {
  u is the Visitor.
  p is the ViewedPage.
} |= Score {
  Visitor is the SegmentedUser.
  Topic from p is the AssignedSegment.
}
```

As a next step, the highest ranked segments for a user have to be found. In this article, we introduce a threshold for the rank. Segments, which have a score above that threshold, apply to a user.

The definitions in Figure 16 drive the selection process.

The highest ranked segments of a user are evaluated inside the concept **SegmentDetermination**, that serves as a scope for executions. The concept **SegmentsOfUser** acts as a function from **User**s to segments with scores above the threshold. That function is invoked within the scope.

The evaluation is based on an **InitialThreshold** that is set inside **SegmentDetermination**. It is set to the value that has to be reached by scored segments.

The first "invocation" of **SegmentsOfUser** for a user collects all **Score**s of the given user. These scores are then narrowed down during function evaluation. Each iteration of the evaluation starts through the concept **Score_rec** that sets the **Value** of the **Score** and decreases both **Value** and **Threshold** by one.

If the **Threshold** reaches **0**, then the score is narrowed down to **IncludedScore**. In that case, the value was greater than the threshold. The score is replaced with the segment in this case, thus terminating the recursion.

If the **Value** reaches **0** first, however, then the value was less than the threshold. In this case the recursion ends without a specific result by replacing the result with **Segments** as declared above.

With these definitions, the retrieval of, for example, segments of a user **Anon815** for threshold **3** is performed as follows:

```
Anon815sSegments is a SegmentDetermination {
  3 is the InitialThreshold.
  SegmentsOfUser {
    Anon815 is the UserToSegment.
  }
}
SegmentsOfUser from Anon815sSegments.
```

As an example for the application of **SegmentDetermination**, the concept **Anon815sSegments** is evaluated and projected to its content **SegmentsOfUser**, the actual "function".

With the track records from the example in Section IX.B above, **Anon815sSegments** evaluates to (equivalent to that example):

```
{ Anon815-on-SportsOverviewPage-1,
  Anon815-on-CulturePage-1,
  Anon815-on-SportsOverviewPage-2 }.
```

This set of concepts constitutes the projection of **SegmentDetermination** to **SegmentsOfUser** and, therefore, further evaluates to **Sports**: Evaluation is applied element-wise, **CulturePage** was not redefined to a **SegmentingWebpage** in our example (thus has no **Topic**), and the other two concepts both evaluate to **Sports**.

XI. APPLYING TARGETING RULES

When users are segmented, the segmentation can be used to create personalized web pages for users.

A. Basic Targeting Execution

Using the model laid out in this paper, web pages are targeted by evaluating a **TargetableWebPage** in the context of the segment(s) of the requesting user. No further modeling means need to be introduced.

More precisely, the syntactical rule of a requested web page is used to produce a document for a **WebPage** concept. In the course of the application of the syntactical rule for the creation of the external representation, all concepts that are referenced in the syntactical rule are evaluated.

In case a user is assigned to more than one segment, one of the segments has to be chosen as the context of the web page evaluation. This has to be done by the environment that handles input and output recognizer and producer in the architecture (see Section IV), and is not presented here.

B. An Example of Web Page Targeting

With the sample concepts defined so far, the statement shown in Figure 17 targets the web page **CulturePage** to user **Anon815**.

```
AssignedSegment
 from SegmentsOfUser from Anon815sSegments {
  CulturePage.
}
```

Figure 17. An example of web page targeting.

As shown above, the segment of user **Anon815** is **Sports** in our running example. Therefore, when the page **CulturePage** is requested by that user in our example, it will be evaluated in the context **Sports**. In this context, the page will have **SoccerExhibition** and **RunningGameScore** as its **NewsTeasers**.

At the same time as the targeted web page is derived, a request for a web page may also increment the matching score as defined in the previous subsection (if it is a **SegmentingWebPage**). This concludes the circle of segmenting and targeting.

This example just demonstrates the selection of content to display at a given position in a web page, as it is also possible with commercial products. With the approach demonstrated here, however, it will also be possible to personalize other aspects of a web page in future work.

XII. SUMMARY AND OUTLOOK

The paper concludes with a summary of this article and an outlook on future research directions.

A. Summary

Forms of personalization are discussed in the literature for quite some time now. Still, integrated models covering most or all aspects of personalization are missing in practice.

This paper presents a study on such an integrated model, that combines content modeling and content rendering with personalization, and that allows expressing various forms of personalization.

The initial modeling approach achieves the goal of integrating content, content representation, users, page visits, segments, user segmentation, and targeting "rules". This integration allows coherent definitions of targeted web sites.

We showed this for predefined segments and a segmentation that is derived from the topics of pages and users visiting those pages.

Technological dependencies were largely avoided. Only for tracking we rely on some additional logic that issues M3L statements based on web requests.

B. Outlook.

This paper concentrates on implicit personalization of presentations for groups of users, in practice called targeting.

Future research will investigate how to employ such integrated models to cover a wider range of personalization approaches and applications. With the help of such models it will be possible to use more than the set of predefined configuration options that contemporary systems exhibit. Instead, these models are expected to unveil personalization capabilities that range over all aspects of services, their content, and their appearance, as well as to give the possibility of utilizing the interconnections between these.

A first step would be to extend the model to other forms of personalization in order to investigate whether these fit in equally well and can be combined within one model.

Content delivery and consumption depend on the context of the user. The utilization of context information for personalization should fit the models well using the M3L. Still, this needs to be studied.

This paper covers an analysis based on a hypothetical model only. It now needs to be connected to a working web server in order to gain practical results.

To increase practical relevance, further information on users should be integrated into the targeting process. Besides the segments derived from user behavior, additional parameters of the context of the user can be taken into consideration.

Contemporary systems incorporate base data of website visitors that have explicitly been raised using other channels, e.g., previous visits or interactions on other channels. Such information may come from a Customer Relationship Management (CRM) system, from transaction processing systems like shop solutions, and from customer journeys.

In practice, the whole information on users may be centralized in a Customer Data Platform (CDP) that contains both the explicit data from the CRM as well as implicit information coming from tracking and segmentation. With such a CDP, targeting is applied to a higher degree than the personalization of single web pages, e.g., by contributing to omnichannel orchestration and customer journey orchestration.

For the examples in this article, it is assumed that a web server is incorporated into systems, and that it can be extended in a way that it analyzes HTTP requests and translates them to some M3L statements. For an even more coherent integrated model of targeting, and in order to avoid the need of additional mapping code outside the models, the HTTP protocol should be modeled in the M3L. E.g., a syntactical rule

```
WebPage |- "GET /" the name "HTTP/1.1".
```

(assuming the concepts with the names in quotes are defined) maps HTTP requests to concepts. Instead of just producing HTML as shown above, further syntactical rules can produce full HTTP responses. This extension of the integrated model has not yet been done.

ACKNOWLEDGMENT

Targeting is one building block in many digitization projects. Valuable insights have, therefore, been gained thanks to exchange with colleagues, partners, and customers.

Though the model presented in this paper is not directly related to the concrete project work at Namics, the author is thankful to his employer for letting him follow his research ambitions based on experience made in customer projects.

REFERENCES

[1] H.-W. Sehring, "An Integrated Model for Content Management, Presentation, and Targeting," Proc. Eleventh International Conference on Creative Content Technologies, May 2019, pp. 26-31.

[2] K. Riemer and C. Totz, "The Many Faces of Personalization: An integrative economic overview of mass customization and personalization," in The Customer Centric Enterprise, M. M. Tseng and F. T. Piller, Eds. Berlin, Heidelberg: Springer, pp. 35-50, 2003.

[3] J. Blom, "Personalization: a Taxonomy," Proc. CHI EA '00 Extended Abstracts on Human Factors in Computing Systems, Apr. 2000, pp. 313-314.

[4] J. Zhang, "The Perils of Behavior-Based Personalization," Marketing Science, vol. 30, pp. 170-186, Dec. 2011.

[5] J. Yan, N. Liu, G. Wang, W. Zhang, Y. Jiang, and Z. Chen, "How much can behavioral targeting help online advertising?" Proc. 18th International Conference on World Wide Web (WWW '09), April 2009, pp. 261-270.

[6] A. S. Das, M. Datar, A. Garg, and S. Rajaram, "Google news personalization: scalable online collaborative filtering," Proc. 16th International Conference on World Wide Web (WWW '07), May 2007, pp. 271-280.

[7] Y. Chen, D. Pavlov, and J. F. Canny, "Large-scale behavioral targeting," Proc. 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '09), June/July 2009, pp. 209-218.

[8] A. L. Montgomery and M. D. Smith, "Prospects for Personalization on the Internet," Journal of Interactive Marketing, vol. 23, pp. 130-137, Jul. 2008.

[9] H.-W. Sehring, "Content Modeling Based on Concepts in Contexts," Proc. Third International Conference on Creative Content Technologies, Sep. 2011, pp. 18-23.

[10] J. W. Schmidt and H.-W. Sehring, "Conceptual Content Modeling and Management," Perspectives of System Informatics, Lecture Notes in Computer Science, vol. 2890, pp. 469-493, 2003.

[11] H. Beales, "The Value of Behavioral Targeting," Mar. 2010. [Online]. Available from: https://networkadvertising.org/pdfs/Beales_NAI_Study.pdf [retrieved: December, 2019]

[12] P. T. Aquino Junior and L. V. L. Filgueiras, "User modeling with personas," Proc. 2005 Latin American Conference on Human-Computer Interaction (CLIHC '05), Oct. 2005, pp. 277-282.

[13] T. Jiang and A. Tuzhilin, "Improving Personalization Solutions through Optimal Segmentation of Customer Bases," IEEE Transactions on Knowledge and Data Engineering, vol. 21, pp. 305-320, Mar. 2009.

[14] J. R. Mayer and J. C. Mitchell, "Third-Party Web Tracking: Policy and Technology," Proc. 2012 IEEE Symposium on Security and Privacy, May 2012, pp. 413-427.

[15] K. S. Kuppusamy and G. Aghila, "A Model for Web Page Usage Mining Based on Segmentation," International Journal of Computer Science and Information Technologies, vol. 2, pp. 1144-1148, 2011.

[16] Website of the CoreMedia AG. [Online]. Available from: https://www.coremedia.com/ [retrieved: December, 2019]

[17] .NET Daily. How to Use Adaptive Personalization with Sitecore [Online]. Available from: https://dotnetdaily.net/sitecore/how-to-use-adaptive-personalization-sitecore [retrieved: December, 2019]

[18] Adobe Target. Behavioral targeting. [Online]. Available from: https://www.adobe.com/marketing/target/behavioral-targeting.html [retrieved: December, 2019]

[19] M. Gorgoglione, C. Palmisano, and A. Tuzhilin, "Personalization in Context: Does Context Matter When Building Personalized Customer Models?" Proc. 6th IEEE International Conference on Data Mining (ICDM 2006), Dec. 2006, pp. 222-231.

[20] H.-W. Sehring, "Context-aware Storage and Retrieval of Digital Content: Database Model and Schema Considerations for Content Persistence," International Journal on Advances in Software, vol. 11, pp. 311-322, Dec. 2018.

[21] T. J. Parr, R. W. Quong, "ANTLR: A predicated-LL(k) parser generator," Software-Practice and Experience, vol. 25, pp. 789–810, Jan. 1995.

[22] T. J. Parr, "Enforcing Strict Model-View Separation in Template Engines," Proc. 13th International Conference on World Wide Web (WWW '04), May 2004, pp. 224-233.

www.iariajournals.org

International Journal On Advances in Intelligent Systems
issn: 1942-2679

International Journal On Advances in Internet Technology
issn: 1942-2652

International Journal On Advances in Life Sciences
issn: 1942-2660

International Journal On Advances in Networks and Services
issn: 1942-2644

International Journal On Advances in Security
issn: 1942-2636

International Journal On Advances in Software
issn: 1942-2628

International Journal On Advances in Systems and Measurements
issn: 1942-261x

International Journal On Advances in Telecommunications
issn: 1942-2601