Managing Complexity: Towards Intelligent Error-Handling Assistance Trough Interactive Alarm Flood Reduction

Sebastian Büttner, Paul Wunderlich, Mario Heinz, Oliver Niggemann, Carsten Röcker

Managing Complexity:
Towards Intelligent Error-Handling Assistance
Trough Interactive Alarm Flood Reduction

Sebastian Büttner, Paul Wunderlich, Mario Heinz, Oliver Niggemann, and
Carsten Röcker

Institute Industrial IT (inIT), OWL University of Applied Sciences, Lemgo, Germany
{sebastian.buettner, paul.wunderlich, mario.heinz, oliver.niggemann,
carsten.roecker}@hs-owl.de

Abstract. The current trend of integrating machines and factories into cyber-physical systems (CPS) creates an enormous complexity for operators of such systems. Especially the search for the root cause of cascading failures becomes highly time-consuming. Within this paper, we address the question on how to help human users to better and faster understand root causes of such situations. We propose a concept of interactive alarm flood reduction and present the implementation of a first vertical prototype for such a system. We consider this prototype as a first artifact to be discussed by the research community and aim towards an incremental further development of the system in order to support humans in complex error situations.

Keywords: Alarm flood reduction; Machine learning; Assistive system

1 Introduction and Motivation

Currently the landscape of industrial production changes fundamentally. While in the past decades engineering in industrial production focused on the question on how to further automatize industrial production, the current development concentrates on the digitalization and connection of industrial components and whole factories to form so-called cyber-physical systems (CPS) [26]. The formed networks as well as the use of machine learning technologies to create intelligent production components will increase production efficiency and is therefore of huge economic value. A study of Germany's digital association Bitkom forecasts a value-added potential of 79 billion euros in the six major industrial branches of the German economy until 2025 [4]. With the changes of technologies and processes, human interactions with production systems and factories will also be transformed [30].

By creating new networks of industrial components and factories, complexity is induced into the system. This complexity can become an issue in case of errors, faults, failures, or security attacks of a single component that might influence other parts of overall CPS due to cascading failures [25]. While some of the

cascading failures might be avoided by incorporating robustness principles into the design and implementation process of CPS (e.g. [43]), we assume that there will be always failure situations that will require human interventions. However, finding root causes in failure situations requires more effort and becomes costly, if systems are connected. Maintenance work in the new complex context of CPS requires interdisciplinary skills and high competencies, such as knowledge about the technologies, economic and legal factors, social and communicational skills and willingness to learn [33]. Based on these high requirements and the present shortage in highly-skilled workers it would be desirable to have intelligent systems that support humans in the analysis and handling of failures in CPS. With this paper, we want to focus on the question on how to help human users to understand root causes of complex failure situations. More specifically, we address the topic of interactive alarm flood reduction in CPS scenarios, an interactive machine learning (iML) approach to find root causes of complex failure situations. With this paper, we present a first prototype of an assistive system that can help users to find root causes of alarm floods by using machine learning technologies.

The rest of the paper is organized as follows. Section 2 gives an overview on related work in the research areas of assistive systems for error-handling, alarm flood reduction and iML. Section 3 shortly describes the research through design method that is the foundation for our research method. Section 4 proposes our concept of interactive alarm flood reduction. Section 5 describes our prototype in detail, followed by a discussion of the results, a conclusion and outlook in section 6.

2 Related Work

In this section, we will provide an overview of the current research regarding three fields: First, we will present previous assistive systems in the context of error-handling. Second, we will show previous machine learning methods for alarm flood reduction and third, we will present the young research field of iML.

2.1 Assistive Systems for Error-Handling

There has been a lot of research work on the topic of assistive systems to support work processes in the future of industrial production and many overview works and surveys exist, e.g. on how to support users with Augmented Reality (AR) in industry [5, 14]. However, only few proposed systems considered the support in complex error situations and focus mainly on a non-interactive visualization. Folmer et al. [15] present a system called Automatic Alarm Data Analyzer (AADA) that use machine learning algorithms to reduce alarm floods and a three-dimensional visualization to show only the most important information to an operator. In a similar way, Laberge et al. [24] analyzed different alarm summary display designs for the visualization of complex error situations. They compared a list-based presentation with a time-series presentation showing icons

and summarized descriptions. According to their study, the latter presentation led to fewer false user responses. Another work in this domain by Renzhin et al. [31] analyses and visualizes communication data between machines in CPS in various way. By choosing the appropriate visualization, users are empowered to find errors more easily. Furthermore, the system can find deviations from the normal state of the system and reports such errors automatically to the user.

2.2 Alarm Flood Reduction

The topic of alarm flood reduction gained attention from the process industry after incidents like the explosion at the Texaco Refinery in Milford Haven [17]. As a result, the non-profit organization Engineering Equipment & Materials Users' Association (EEMUA) created the guideline EEMUA 191 [10]. The quasi-standard EEMUA 191 for alarm management recommends to have only one alarm per 10 minutes. Based on this, further standards such as ANSI/ISA-18.2-2009 [28] or IEC62682:2014 [9] have been developed. With the vision of industry 4.0 this topic has an increasing importance also for other industrial branches. Several approaches to address the topic of alarm floods exist. Most of them deal with clustering similar alarms together. A good overview about different approaches is provided by Wang et al. [39].

We want to focus on reducing alarm floods by identifying the root cause of the alarm flood. Therefore, we need a causal model which represents the dependencies of the alarms. Probabilistic graphical models, such as Bayesian nets, fault trees, or Petri nets are particularly suitable for this purpose. They were already used in the field of alarm flood reduction. Kezunovic and Guan [23] use a fuzzy reasoning Petri nets diagnose model to identify root causes. For this, they take advantages of an expert rule based system and fuzzy logic. Simeu-Abazi et al. [35] exploit dynamic fault trees to filter false alarms and locate faults from alarms. Guo et al. [16] develop an analytic model for alarm processing, which is based on temporal constraint networks. They use this model to find out what caused the reported alarms and estimate when these events happen. Wei et al. [41] apply a rule network for alarm processing. They can determine the root cause and also identify missing or false alarms. Wang et al. [37] combine association rules with fuzzy logic. They use a weighted fuzzy association rule mining approach to discover correlated alarm sequences. Based on this, they are able to identify root causes. Abele et al. [1] propose to combine modeling knowledge and machine learning knowledge to identify alarm root causes. They use a constrained-based method to learn the causal model of a factory represented by a Bayesian network. This enables faster modeling and accurate parametrization of alarm dependencies but expert knowledge is still required. Wang et al. [38] apply an online root-cause analysis of alarms in discrete Bayesian networks. They restrict the Bayesian Network to have only one child. The method is evaluated on a numerical example of a tank-level system. In [42] various approaches to learn a causal representation of alarms to identify the root cause are investigated. The concept for the reduction of alarm floods in this work is based on their findings.

2.3 Interactive Machine Learning (iML)

In the context of Machine Learning, iML is an approach that involves humans into machine learning processes to deliver better results than fully-automated systems. According to Robert et al. [32], the available iML system can be classified into the following three categories:

First, there are iML systems, where an appropriate data visualization helps humans to manually build a model. Still, the interference is done automatically based on the manually created model. Examples for this system are the perception-based classification (PBC) system presented by Ankerst et al. [2, 3] and the work of Ware et al. [40].

Second, systems have been described under the term iML that involve humans into an evaluation-feedback loop to improve an automatically generated model. The created model is appropriately visualized to the user and the system contains a feedback channel, where users can assess the quality of a certain model. With this feedback, the system can improve the model iteratively. These systems require a very fast model generation to allow rapid improvement when feedback is given to the system. Examples for such systems are the system of Fails et al. [11] that allows interaction designers to rapidly build perceptual user interfaces (PUI) and the Wekinator[1] system that analyses human gestures in the context of music making [12].

The third type that is also named collaborative iML (ciML) is the newest of the approaches. ciML systems can be characterized as system, where humans can influence the model generation algorithm during runtime and work collaboratively with the algorithm to create a model. Examples for ciML systems are the decision tree generation system by Ankerst et al. [3] and interactive ant-colony algorithm to solve the traveling salesman problem (TSP) presented by Holzinger et al. [20].

In this paper, we understand iML as systems, were the user is involved into the feedback loop of iML systems, which is the second type in the classification above.

3 Method

To better understand, how to support users in handling complex failure situations, we decided to develop a first prototype of an interactive system for alarm flood reduction. This prototype can be seen as a research artifact to communicate and discuss ideas among the research community [27]. In this way, we follow the research through design approach proposed by Zimmerman et al. [44]. To develop the prototype, we chose an iterative approach and followed the human-centered design process specified in ISO 9241-210 (see [21], cf. [19]). This process has proven to be very effective for designing interactive systems with a high usability and has been successfully applied in an industrial context as well [6].

---

[1] http://www.wekinator.org/

The prototype presented in this paper is the result of the first iteration. It is a vertical prototype, so it does not contain all features of the system yet, but therefore contains the core functionality and shows the feasibility of our concept.

4 Concept

The general concept proposed in this paper is the use of a cloud-based machine learning system that monitors data from machines and becomes active in the situation of an alarm flood, which we define as an occurrence of more than ten errors per minute. This concept is shown in Figure 1. Data, such as error
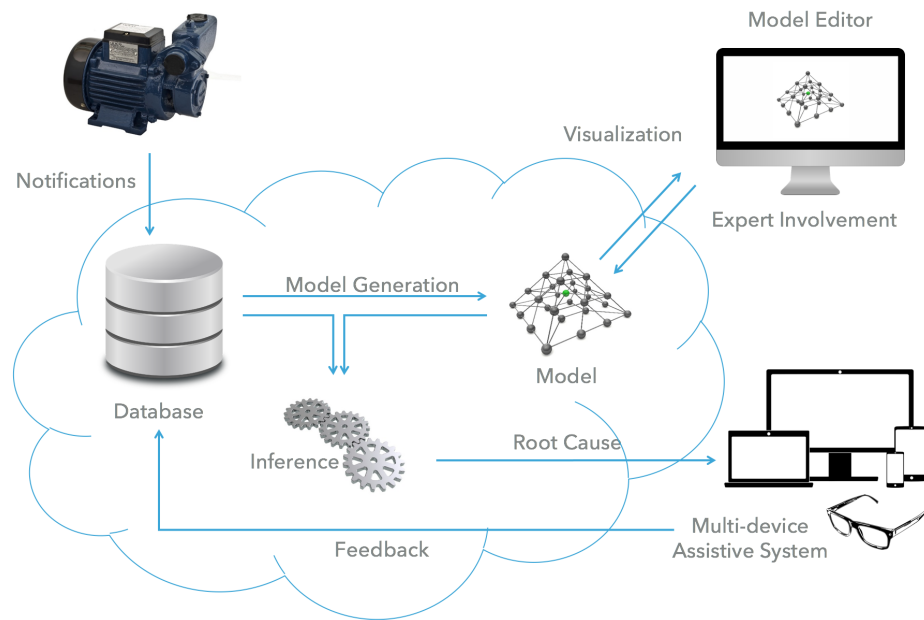

Fig. 1: Overall process of the interactive alarm flood reduction

messages and warnings from machines and factories are send to a cloud service and stored in a database. Based on this data, a machine learning algorithm builds up a model of the relation of different error messages and warnings. In case of an alarm flood the model is used to infer conclusions about the root cause of the current situation. The analysis is presented to the user in a suitable way (e.g. via an adaptive and responsive user interface). There are two types of user involvement in this concept: First, the users of the assistive system (usually the operators of factories or machines) report (explicitly or implicitly) feedback to the the system, if a certain root cause was predicted correctly from the system. This feedback is stored in the database and used for the next iteration of the

model generation. Second, expert users can improve the model by using a model editor. By assessing, if a detected error correlation is a causal relation or not, the expert can manually change the model. The manual changes are kept when the model is rebuilt in following iterations.

This section will focus on the aspects of the conceptual development of the alarm flood reduction (algorithm) and the adaptive and responsive user interface.

4.1 Alarm Flood Reduction

The concept of the alarm flood reduction is depicted in Figure 2. The concept can be divided into two steps. As an initial step a causal model from the gathered data in the alarm log is learned. In the alarm log the alarms of the factory or CPS are listed with information about timing, description and status. The status represents if the alarm is active or inactive. Based on the information of the alarm log a causal model of the alarms is learned. Probabilistic graphical models are suitable for as a causal model. We decide to use Bayesian Networks to represent the dependencies of the alarms as a causal model. Bayesian networks are a class of graphical models which allow an intuitive representation of multivariate data. A Bayesian network is a directed acyclic graph, denoted $B = (N, E)$, with a set of variables $\boldsymbol{X} = \{X_1, X_2, \ldots, X_p\}$. Each node $n \in N$ is associated with one variable X_i. The edges $e \in E$, which connect the nodes, represent direct probabilistic dependencies. In a second step, we use the current alarms and the learned causal model of the alarms to infer the root cause. Because of the learned relations, we are able to identify the possible root cause of an alarm flood. As a consequence, we can reduce the amount of alarms to the possible root cause.
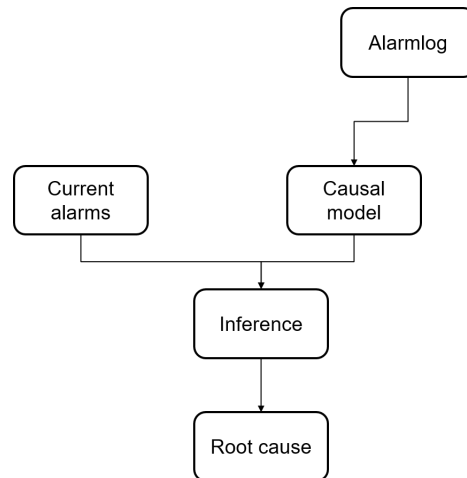
Fig. 2: Concept of alarm flood reduction

4.2 Adaptive and Responsive User Interface

We distinguish between two separate user interfaces: the interface of the assistive system and the model generator.

The user interface of the assistive system should be highly flexible to support different users and work situations. Therefore, we propose an adaptive and responsive user interface, that can be used with various technologies and interaction concepts. As a basic user interface, we use a responsive web application that adapts to different devices. While responsive web applications are the current state of web technology, they are limited to (two-dimensional) graphical user interfaces. With this interaction concept, we go beyond this limitation. The web application is the base visualization; however, if a user needs further assistance, such as support in a repair scenario, where free hands are required, the user can switch to an augmented reality (AR) visualization based on head-mounted displays (HMD) (such as [29]) or mobile in-situ projections (such as [7]). Beyond the visualization, the user needs to have a way of giving feedback to the system, while and after doing maintenance work. Especially the results of an obtained repair task are important to check, whether the model is accurate or need to be improved. Therefore, we included a feedback channel to the system to report, whether the root cause was detected correctly. This feedback channel could be explicitly by users approvals or implicitly (cf. [34]) by the recognition of the users activities and tasks.

The model generator is used by experts to manually adapt the model. This interface presents the model in the form of a large graph. Due to the large data sets that might be involved, this interface is shown on large computer displays. Expert users can mark edges that represent causalities of errors or remove edges from the model, that are no causal relations.

5 Prototype

In the context of the human-centered design process (compare section Method), a first prototype has been developed iteratively. This prototype represents a vertical prototype, so it does not contain all features of the system yet. However, it contains the core functionality and shows the feasibility of our proposed concept. This section will describe the implementation of the prototype system.

5.1 Architecture

The high-level architecture of the presented system is shown in Figure 3. For the basic infrastructure, we use Amazon Web Services (AWS), which provides services, such as virtual servers, a database (DynamoDB), and supports the development of application programming interfaces (APIs) that follow the principles of representational state transfer (RESTful APIs). We provide a web server with a generic RESTful API, where different machines can be connected to. Since industrial machines usually do not use web protocols but rather rely on

Machine / Component # 1

Machine-specific Connector

Machine / Component # ...

Machine-specific Connector

Machine / Component # n

Machine-specific Connector

Webserver with RESTful API

Database

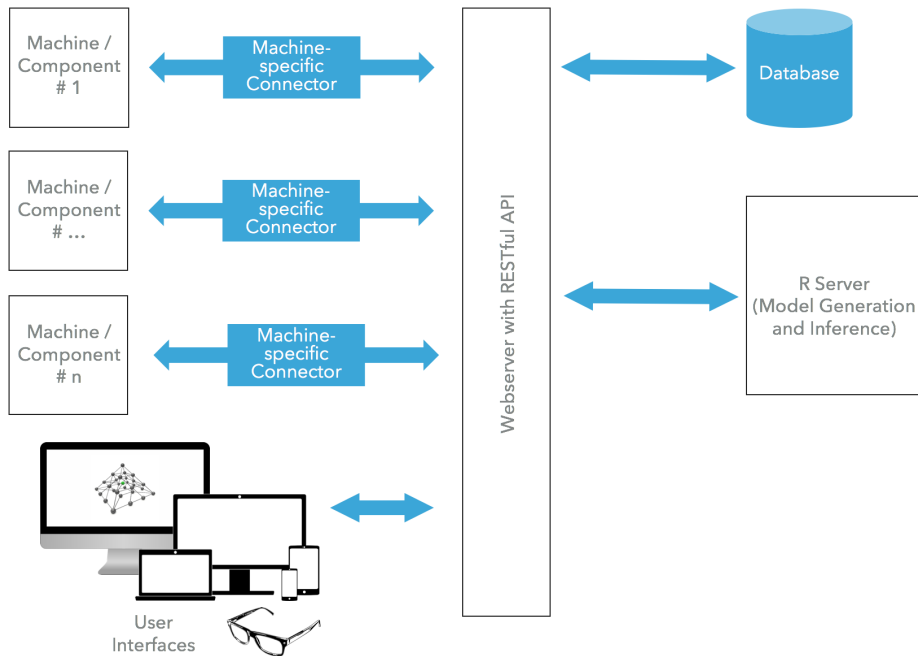R Server (Model Generation and Inference)

User Interfaces

Fig. 3: High-level architecture of the system

machine to machine protocols, such as Open Platform Communications Unified Architecture (OPC UA), individual connectors need to be implemented for each of the specific machine types. The web server is connected to the AWS DynamoDB and to a server running an R environment[2] for executing the machine learning algorithms. The web server is also used to provide the current user interface to various devices. The following sections present the implementation of the machine learning algorithms and of the user interface in more detail.

5.2 Machine Learning Algorithms

The implementation of the machine learning algorithm contains three steps, namely structure learning, parameter learning and inference. We use the Max-Min Hill-Climbing which was developed by Tsamardinos et al. for learning the structure [36]. For a better understanding of the associated pseudo code, we need a few definitions. The dataset D consists of a set of variables ϑ. In the variable PC_x the candidates of parents and children for the node X are stored. This set of candidates is calculated with a Max-Min Parents and Children (MMPC) algorithm. The variable Y is a node of the set PC_x. The pseudo code of MMHC looks as follows: The algorithm first identifies the parents and children set of each variable, then performs a greedy Hill-Climbing search in the reduced space

---

[2] https://www.r-project.org/

Algorithm 1 MMHC Algorithm

 1: **procedure** MMHC(D)
 2:     Input: data D
 3:     Output: a DAG on the variables in D
 4:     % Restrict
 5:     **for** every variable $X \in \vartheta$ **do**
 6:         $PC_X = \text{MMPC}(X, D)$
 7:     **end for**
 8:     % Search
 9:     Starting from an empty graph perform Greedy Hill-Climbing with operators add-edge, delete-edge, reverse-edge. $Y \to X$ if $Y \in PC_X$
10:      **Return** the highest scoring DAG found
11: **end procedure**

---

of Bayesian network. The search begins with an empty graph. The edge addition, removal, or reversing which leads to the largest increase in the score is taken and the search continues in a similar way recursively. The difference from standard Hill-Climbing is that the search is constrained to only consider edges which were discovered by MMPC in the first phase. The MMPC algorithm calculates the correlation between the nodes.

In a next step, the probabilities or parameters of the Bayesian network are learned. For this we use the maximum likelihood estimation (MLE) which was developed by R. A. Fischer and is a classical method in statistics [13]. Here, a parameter p is estimated to maximize the probability of obtaining the observation under the condition of the parameter p. In other words, the MLE provides the most plausible parameter p as an estimate with respect to the observation. If the parameter p is a probability in the Bayesian network and the historical data D represents the observations, the likelihood function is composed as follows:

$$L(D|p) = \prod_{i=1}^{n} f(D|p) \tag{1}$$

The probability density function of D under the condition p is $f(D|p)$. With the learned structure and the probabilities, the inference can begin. The approximate method logic sampling (LS) is used for this purpose.

The Logic Sampling Algorithm is a very simple procedure developed by Max Henrion in 1986 [18]. In this case, a state is arbitrarily assumed per sample for the root nodes according to their probability table. Thus, a certain number of samples, which are determined, are carried out. Subsequently, the probability that e.g. a node X assumes the state True as follows:

$$P(X = True) = \frac{\text{Number of cases with X=True}}{\text{Number of all samples}} \tag{2}$$

This process always converges to the correct solution, but in very rare cases the number of samples required can become exorbitant [22]. The feedback of the expert is included with a whitelist and blacklist in the structure learning. The whitelist contains all predetermined relations and the blacklist contains all prohibited relations between the alarms.

5.3 User Interface

The current user interface is built on the basis of web technologies. Server-sided, the system provides a generic RESTful API for exchanging the data between the server and the end-users' devices. The actual client is built as a responsive web application. While the responsive assistive system is part of the current prototype, the model editor is ongoing work and therefore not further described in this section.
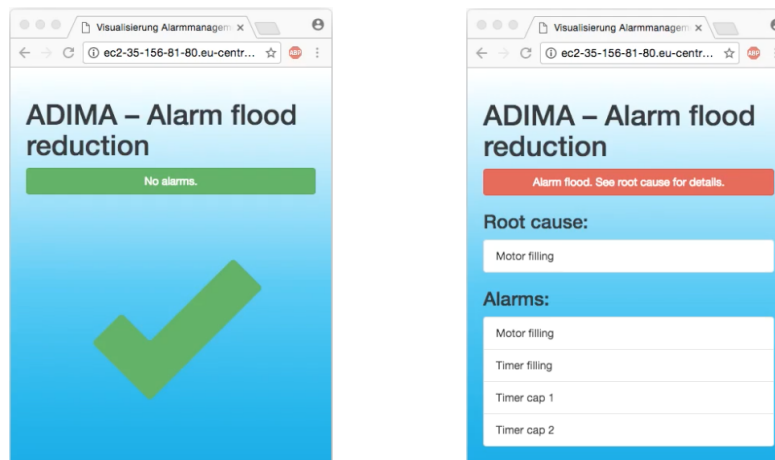

Fig. 4: Screenshots of the responsive web application with (a) normal situation without errors and (b) case of an occurring alarm flood.

The webserver can be accessed to load the assistive system in form of a responsive web application that automatically adapts to the device of the user, such as computer screens, smartphones or tablet computers. The integration of other interaction devices, such as HMDs for hand-free operation is ongoing work. The user interface is implemented in HTML5 using the frameworks Bootstrap[3] and AngularJS[4]. To display the information on small screens, the content of the current user interface is limited to the most important information. It shows the occurring errors and (in case of an alarm flood) the root cause of the problem

---

[3] http://getbootstrap.com/
[4] https://angularjs.org/

(see Figure 4 b) or a huge green check mark if all machines are working properly (see Figure 4 a).

Since this responsive website cannot be displayed on devices without web browsers, such as HMDs, native applications will be provided in future that also make use of the generic API. According to our concept, there will be more detailed assistance information to display not only the root cause but rather various types of media (images, videos) that help users to fix a specific cascade of errors, which is a topic for future work.

6 Discussion, Conclusion and Outlook

In this paper, we proposed our concept of a system for interactive alarm flood reduction for determine root causes in complex failure situations. Such a system will be beneficial for finding errors in complex industrial environments, e.g. in the future of production, when CPS become omnipresent. Contrary to previous work, we do not only focus on automatic machine learning concepts for alarm flood reduction, but rather want to establish a feedback channel to empower users to give feedback of the results of the alarm flood reduction to the system. Having this possibility, the model representing the causal relations of alarms can iteratively be improved.

We presented a first vertical prototype of our concept consisting of a machine learning system based on the Max-Min Hill-Climbing algorithm of Tsamardinos et al. [36] and the maximum likelihood estimation of Fischer [13] as well as a responsive web application as user interface based on web technology. This prototype has been developed in an iterative design process. While a formal evaluation of our prototype is outstanding and future work, we discussed the results with experts from industry. However, the prototype shows the feasibility of our proposed concept.

Future work will focus on the improvement of the system to give better predictions of root causes as well as a deeper integration of human feedback. Furthermore, additional features will be added to the very basic user interface described in this paper. The responsive web application will be extended in the way to actually realize an assistive system; this will be achieved by not only showing the error but rather guide users through the process of repair or maintenance in a similar way to assistive systems in manufacturing, such as [8]. For this purpose, the system will be extended to also allow for connecting other interaction devices, e.g. HMDs. While the interface of the web server is designed in a way to cover this process, a specific scenario containing machines and related maintenance and repair manuals needs to be created. Parallel to this technical development, we plan to evaluate the overall systems repeatedly during the iterative development process.

As already stated our system is in an early stage. However, the current prototype is a first artifact to communicate our vision and to discuss ideas among the research community (cf. [27, 44]). The further development and evaluation will

give more insight into the question on how to help human users to understand errors in complex interconnected systems, such as CPS.

Acknowledgments

References

1. L. Abele, M. Anic, T. Gutmann, J. Folmer, M. Kleinsteuber, and B. Vogel-Heuser. Combining knowledge modeling and machine learning for alarm root cause analysis. *IFAC Proceedings Volumes*, 46(9):1843 – 1848, 2013.
2. M. Ankerst, C. Elsen, M. Ester, and H.-P. Kriegel. Visual classification: an interactive approach to decision tree construction. In *Proceedings of the fifth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 392–396. ACM, 1999.
3. M. Ankerst, M. Ester, and H.-P. Kriegel. Towards an effective cooperation of the user and the computer for classification. In *Proceedings of the Sixth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '00, pages 179–188, New York, NY, USA, 2000. ACM.
4. W. Bauer and P. Horváth. Industrie 4.0 – Volkswirtschaftliches Potenzial für Deutschland. *Controlling*, 27(8-9):515–517, 2015.
5. S. Büttner, H. Mucha, M. Funk, T. Kosch, M. Aehnelt, S. Robert, and C. Röcker. The design space of augmented and virtual reality applications for assistive environments in manufacturing: A visual approach.
6. S. Büttner and C. Röcker. Applying human-centered design methods in industry – a field report. In *Human-Computer Interaction–Perspectives on Industry 4.0. Workshop at i-KNOW 2016 Graz, Austria*, 2016.
7. S. Büttner, O. Sand, and C. Röcker. Extending the design space in industrial manufacturing through mobile projection. In *Proceedings of the 17th International Conference on Human-Computer Interaction with Mobile Devices and Services Adjunct*, pages 1130–1133. ACM, 2015.
8. S. Büttner, O. Sand, and C. Röcker. Exploring design opportunities for intelligent worker assistance: A new approach using projection-based ar and a novel handtracking algorithm. In *European Conference on Ambient Intelligence*, pages 33–45. Springer, 2017.
9. I. E. Commission. En-iec 62682:2014 management of alarm systems for the process industries. Technical report, International Electrotechnical Commission.
10. E. Equipment, M. U. Association, E. Equipment, and M. U. A. Staff. *Alarm Systems: A Guide to Design, Management and Procurement*. EEMUA publication. EEMUA (Engineering Equipment & Materials Users Association), 2007.
11. J. A. Fails and D. R. Olsen Jr. Interactive machine learning. In *Proceedings of the 8th international conference on Intelligent user interfaces*, pages 39–45. ACM, 2003.
12. R. Fiebrink, P. R. Cook, and D. Trueman. Human model evaluation in interactive supervised learning. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '11, pages 147–156, New York, NY, USA, 2011. ACM.

13. R. A. Fisher. On an absolute criterion for fitting frequency curves. *Messenger of Mathematics*, 41:155–160, 1912.

14. P. Fite-Georgel. Is there a reality in industrial augmented reality? In *Mixed and Augmented Reality (ISMAR), 2011 10th IEEE International Symposium on*, pages 201–210. IEEE, 2011.

15. J. Folmer, D. Pantförder, and B. Vogel-Heuser. An analytical alarm flood reduction to reduce operators workload. *Human-Computer Interaction. Users and Applications*, pages 297–306, 2011.

16. W. Guo, F. Wen, Z. Liao, L. Wei, and J. Xin. An analytic model-based approach for power system alarm processing employing temporal constraint network. *IEEE Transactions on Power Delivery*, 25(4):2435–2447, Oct 2010.

17. G. B. Health and S. Executive. *The Explosion and Fires at the Texaco Refinery, Milford Haven, 24 July 1994: A Report of the Investigation by the Health and Safety Executive Into the Explosion and Fires on the Pembroke Cracking Company Plant at the Texaco Refinery, Milford Haven on 24 July 1994*. Incident Report Series. HSE Books, 1997.

18. M. Henrion. Propagating uncertainty in bayesian networks by probabilistic logic sampling. In *Uncertainty in Artificial Intelligence 2 Annual Conference on Uncertainty in Artificial Intelligence (UAI-86)*, pages 149–163, Amsterdam, NL, 1986. Elsevier Science.

19. A. Holzinger, M. Errath, G. Searle, B. Thurnher, and W. Slany. From extreme programming and usability engineering to extreme usability in software engineering education. In *Computer Software and Applications Conference, 2005. COMPSAC 2005. 29th Annual International*, volume 2, pages 169–172. IEEE, 2005.

20. A. Holzinger, M. Plass, K. Holzinger, G. C. Crisan, C.-M. Pintea, and V. Palade. Towards interactive machine learning (iML): Applying ant colony algorithms to solve the traveling salesman problem with the human-in-the-loop approach. In *IFIP International Cross Domain Conference and Workshop (CD-ARES), page in print. Springer, Heidelberg, Berlin, New York*, pages 81–95, 2016.

21. ISO. 9241-210: 2010. Ergonomics of human system interaction – part 210: Human-centred design for interactive systems. *International Standardization Organization (ISO). Switzerland*, 2009.

22. L. Kanal and J. Lemmer. *Uncertainty in Artificial Intelligence 2*. Machine Intelligence and Pattern Recognition. Elsevier Science, 2014.

23. M. Kezunovic and Y. Guan. Intelligent alarm processing: From data intensive to information rich. In *2009 42nd Hawaii International Conference on System Sciences*, pages 1–8, Jan 2009.

24. J. C. Laberge, P. Bullemer, M. Tolsma, and C. R. Dal Vernon. Addressing alarm flood situations in the process industries through alarm summary display design and alarm response strategy. *International Journal of Industrial Ergonomics*, 44(3):395–406, 2014.

25. E. A. Lee. Cyber physical systems: Design challenges. In *Object Oriented Real-Time Distributed Computing (ISORC), 2008 11th IEEE International Symposium on*, pages 363–369. IEEE, 2008.

26. J. Lee, B. Bagheri, and H.-A. Kao. A cyber-physical systems architecture for industry 4.0-based manufacturing systems. *Manufacturing Letters*, 3:18–23, 2015.

27. H. Mucha and K. Nebe. Human-centered toolkit design. In *HCITools: Strategies and Best Practices for Designing, Evaluating and Sharing Technical HCI Toolkits, Workshop at CHI'17, Denver, USA*. ACM, 2017.

28. I. S. of Automation and A. N. S. Institute. *ANSI/ISA-18.2-2009, Management of Alarm Systems for the Process Industries*. ISA, 2009.

29. V. Paelke, C. Röcker, N. Koch, H. Flatt, and S. Büttner. User interfaces for cyber-physical systems. *at-Automatisierungstechnik*, 63(10):833–843, 2015.

30. R. R. Rajkumar, I. Lee, L. Sha, and J. Stankovic. Cyber-physical systems: the next computing revolution. In *Proceedings of the 47th Design Automation Conference*, pages 731–736. ACM, 2010.

31. D. Renzhin, D. Pantförder, J. Folmer, and B. Vogel-Heuser. Darstellungskonzepte für die zustandsabhängige Diagnose industrieller Kommunikationsnetzwerken für verteilte Automatisierungssysteme. In *MBEES*, pages 91–100, 2012.

32. S. Robert, S. Büttner, C. Röcker, and A. Holzinger. Reasoning under uncertainty: Towards collaborative interactive machine learning. In *Machine Learning for Health Informatics*, pages 357–376. Springer, 2016.

33. M. Schenk. *Instandhaltung technischer Systeme: Methoden und Werkzeuge zur Gewährleistung eines sicheren und wirtschaftlichen Anlagenbetriebs.* Springer-Verlag, 2009.

34. A. Schmidt. Implicit human computer interaction through context. *Personal technologies*, 4(2):191–199, 2000.

35. Z. Simeu-Abazi, A. Lefebvre, and J.-P. Derain. A methodology of alarm filtering using dynamic fault tree. *Reliability Engineering & System Safety*, 96(2):257 – 266, 2011.

36. I. Tsamardinos, L. E. Brown, and C. F. Aliferis. The max-min hill-climbing bayesian network structure learning algorithm. *Machine Learning*, 65(1):31–78, 2006.

37. J. Wang, H. Li, J. Huang, and C. Su. Association rules mining based analysis of consequential alarm sequences in chemical processes. *Journal of Loss Prevention in the Process Industries*, 41:178 – 185, 2016.

38. J. Wang, J. Xu, and D. Zhu. Online root-cause analysis of alarms in discrete bayesian networks with known structures. In *Proceeding of the 11th World Congress on Intelligent Control and Automation*, pages 467–472, June 2014.

39. J. Wang, F. Yang, T. Chen, and S. L. Shah. An overview of industrial alarm systems: Main causes for alarm overloading, research status, and open problems. *IEEE Transactions on Automation Science and Engineering*, 13(2):1045–1061, April 2016.

40. M. Ware, E. Frank, G. Holmes, M. Hall, and I. H. Witten. Interactive machine learning: letting users build classifiers. *International Journal of Human-Computer Studies*, 55(3):281–292, 2001.

41. L. Wei, W. Guo, F. Wen, G. Ledwich, Z. Liao, and J. Xin. An online intelligent alarm-processing system for digital substations. *IEEE Transactions on Power Delivery*, 26(3):1615–1624, July 2011.

42. P. Wunderlich and O. Niggemann. Structure learning methods for bayesian networks to reduce alarm floods by identifying the root cause. In *22nd IEEE International Conference on Emerging Technologies and Factory Automation (ETFA 2017)*, Sep 2017.

43. O. Yagan, D. Qian, J. Zhang, and D. Cochran. Optimal allocation of interconnecting links in cyber-physical systems: Interdependence, cascading failures, and robustness. *IEEE Transactions on Parallel and Distributed Systems*, 23(9):1708–1720, 2012.

44. J. Zimmerman, J. Forlizzi, and S. Evenson. Research through design as a method for interaction design research in hci. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 493–502. ACM, 2007.