Agent-based assistance system for the dynamic reliability calculation of cyber-physical systems

Nasser Jazdi, Philipp Marks
Institute of Industrial Automation and Software Engineering
University of Stuttgart, Germany.
{nasser.jazdi, philipp.marks}@ias.uni-stuttgart.de

Abstract— **The concept of cyber-physical systems introduces new technologies that enable new kinds of functionalities. At the same time, these technologies also pose new requirements for the reliability calculation of such systems. Runtime data of the system can be made available worldwide through the networking of components. This offers the possibility to use that data for a more accurate dynamic reliability calculation. Another aspect to consider in reliability calculation of CPS is that the system structure is not yet known at the time of system development and is likely to change frequently over time. This paper presents a multi-agent-based assistance concept for dynamic reliability calculation of cyber-physical systems. The failure rate which is usually assumed to be constant, is divided into time-dependent partial failure rates which are calculated dynamically by the agents. To perform this calculation, the agents gather relevant information from various sources and collaborate with each other. The paper also shows the influence of dynamic factors on the reliability of a system using a software prototype.**

Keywords—dynamic reliability calculation, cyber-physical systems, multi agent systems, assistance systems

I. INTRODUCTION

Automated systems permeate our everyday lives, and can be found in all sorts of devices. The development from simple devices to "smart devices" has long been in full swing. For example, while first simple mobile phones could only make phone calls and send text messages, a smartphone nowadays can be used to surf the internet and to perform various actions for which a computer was previously needed. Through cloud computing and the entry of the internet into industrial production, task distribution no longer works via a central control unit, but decentrally via "autonomous, powerful microcomputers" which communicate wirelessly or via the internet. At this point, the physical world and the virtual world are moving closer together. At the intersection of these two worlds "cyber-physical systems" (CPS) are located. Some of the aspects covered by CPS in production are: intelligent machines, storage systems from equipment to logistics, marketing and production per se [1].

CPS consist of hardware and software units that contain various sensors and actuators and can also process and store data or communicate via networks. Data and services can be used worldwide. In addition, a human machine interface is provided. As a result, traditional controllers, such as programmable logic controllers, can be replaced [2].

These changes and capabilities enable new approaches towards automation such as distributed and dynamic production. Automation units become service providers in the production network and customers can create their own solution using these services, which are optimized in terms of time, reliability, costs, etc. Constant monitoring of production resources worldwide has become possible which enables to react quickly and efficiently on market changes. There are completely new possibilities for diagnosis and maintenance. On the other hand, these changes bring about new challenges. In automation technology, specific constraints such as real-time requirements, scarce computing power and resources play an essential role and make these systems special cases. We need new approaches and concepts for calculating the reliability, testing, diagnostics, maintenance and control of these applications.

The constant availability of data from individual automation components paired with cloud computing and cyber technologies enables new approaches to the dynamic calculation of reliability. This makes the individualized maintenance of automated systems possible and recall actions can be reduced or even prevented. The paradigm of multi agent systems can be applied to gather reliability-relevant data and for calculating the reliability of CPS dynamically. Through agent collaboration, the dynamic influence parameters on the reliability of the overall system can be determined at runtime of the system.

The rest of the paper is structured as follows: Section II describes the status quo of reliability calculation and explains why assistance in dynamic reliability calculation is needed. Subsequently, applications of agents in assistance systems are reviewed in Section III. Section IV presents the approach for the dynamic reliability calculation and the concept of a multi agent system architecture for that purpose. To visualize the effects of dynamic influence parameters on reliability, the results of a simulated scenario are shown in Section V. Finally, in Section VI the paper closes with conclusions and an outlook on future work.

II. RELIABILITY CALCULATION -STATUS QUO-

Reliability is the probability that a system fulfils the required specified function within a certain time interval, and under permissible operating conditions [3]. Measures in reliability

978-1-5386-2205-6/18/$31.00 ©2018 IEEE

engineering serve to prevent errors or failures. These measures are implemented in order to maintain profitability and to meet warranty periods [4]. Reliability is influenced by factors such as increased system complexity, greater functionality, increased customer demands, increasing product liability etc. Basically, there are two types of methods to calculate the reliability of a system. If the empirical data on the failures of the system to be considered is available, it is possible to calculate the failure rate empirically and then derive the reliability *R(t)* from it. Usually, the failure rate is considered as constant, which may not be accurate when it comes to dynamic systems like CPS. However, this method can only be used in systems which are not very complex. For larger systems, the proof of reliability - with regard to dangerous states - can be carried out analytically [4] [5]. Thus, assistance is needed in this process.

In the following, this article describes an assistance system for the dynamic reliability calculation, taking different influential factors into account. The assistance system consists of two parts: the agent-based part, which encapsulates the functionality of the calculation algorithms, and realizes the communication between subsystems and data acquisition, as well as the actual reliability calculation.

III. Application of the Agent Paradigm for the Realisation of Assistance Systems

The software agent paradigm has been researched and developed since the 1990s in a variety of different disciplines. Numerous applications of multi agent systems in the field of industrial automation have emerged, which are used at all stages of the automated system life cycle.

Although no universally accepted definition of agents can be found, there is still consensus about the typical characteristics that make up an agent. In addition to the autonomy of its behavior, this includes the goal orientation in particular. In VDI Guideline 2653, agents are defined as follows: "A technical agent is a delimited (hardware or/and software) entity with defined goals. A technical agent strives to achieve these goals through autonomous behavior, interacting with his environment and with other agents." [16]. The following considerations are based on this definition. The proposed concept uses agents that appear as a pure software unit as well as agents that represent an individual component consisting of hardware and software.

Different research directions can be identified which deal with the issue of agents in the field of automation technology. For example, agents are used to control intralogistics systems [6] or to control continuous processes [7]. Self-management approaches were explored within the operational phase of automated systems, with special emphasis on the autonomy properties of agents [8]. Numerous application examples can be found not only in the operational phase of automated systems, but also in the earlier lifecycle phases, such as the design phase. In [9] an agent system is described which performs the consistency check of mechatronic models by raising the heterogeneous models of different disciplines onto a global, cross-model level of abstraction, where they are then examined. Wagner [10] describes how to use agents to actively assist the engineer during the use of computer-aided tools (CAx). By using stored engineering knowledge and known dependencies between components and modules, the system recognizes interdependencies, as well as problems in the current design and proposes possible solutions proactively to the user.

The two last-mentioned approaches can be assigned to the category of assistance systems, in which further application examples for agents can be found. The goal of assistance systems is to offer the user context-sensitive support, for example by providing useful information, or by performing supporting activities. In [11], an assistance system is described which uses agents to carry out the rough planning for automated systems and their subsequent adaptation to changed boundary conditions. Furthermore, agent approaches were also successfully implemented in resource planning, for example in the scheduling of airplanes [12] or the planning and optimization of logistics in production processes [13]. Cavalieri et al. [14] describe an agent-based system for predictive maintenance planning. An overview of implemented agent concepts in the field of intelligent energy systems can be found in [15]. The numerous examples show that agents can be used profitably in different scenarios and are particularly suitable for the implementation of assistance systems.

The aforementioned agent properties can be used profitably in the dynamic calculation of the reliability of an automated system. The trend towards distributed automation systems, in which individual components function as a self-contained unit, can be implemented in analogy to reality through the agents' encapsulation concept. The autonomy of the agents makes it possible to recognize and take the dynamic factors into account when calculating the reliability of the system during runtime. By interacting with each other, the different types of agents can achieve the overall goal of the system. The approach for the dynamic reliability calculation is shown in the following section. Subsequently, the proposed approach is integrated into an agent-based assistance system.

IV. Approach for the Dynamic Calculation of Reliability

Implementing a cyber-physical system concept has a major impact on the reliability of systems. Among other things, this results in a higher level of complexity, a wider functionality which is therefore more complex to maintain and increased customer requirements for quality, functionality and usability. These requirements complicate product development and represent a risk for a high level of reliability and availability. Even with standardized methods of reliability analysis and supporting software tools, it is hardly possible to consider an entire system at once. The methods are instead applied to individual modules. It is not always easy to make an adequate separation between them, as there are interdependencies between different subsystems. These interdependencies, as well as consequential errors and temporal dependencies, are difficult or even impossible to analyze using the conventional methods. This is especially problematic for a detailed software reliability analysis.

The high degree of interconnectedness that underpins the fourth industrial revolution not only holds risks for data, but also for reliability. Effects on other participants cannot always be predicted when developing components. Due to the Internet of

Things, there is an increasing amount of interaction, communication and related counter-effects. An error can spread easily and quickly, leading to unwanted secondary outages in different places if they are information-dependent from each other. In the process, it is difficult to trace events that had not been predicted when developing the component. Failure of one production site in a distributed automated production system can also endanger other sites. It makes sense for the manufacturer and the customers to be aware of the probability of failure of the system or individual modules as precisely as possible. Even if it is quick and easy to repair a system, repair is always expensive in terms of time and money. Depending on the system, a malfunction may also involve a failure of other important functions. If the average time between two malfunctions is known, maintenance intervals can be optimized. The occurrence of many failures can be avoided with minor maintenance effort. Due to the communication ability of "things" in the Internet of Things, repair costs can also be reduced by automatically sending information about the malfunction to the responsible personnel after the malfunction has occurred.

For large systems, which are composed of many different components from different manufacturers and domains, it is difficult to determine an exact failure rate during product development and to therefore make reliable statements about the system availability. Therefore, a concept is needed that gathers, weighs and processes qualitative and quantitative information about the system from different sources while the system operates, in order to provide reliability information. Ideally, this is an automatable concept that can easily integrated and process new information. Figure 1 shows the basic structure of such a system for a dynamic reliability calculation.

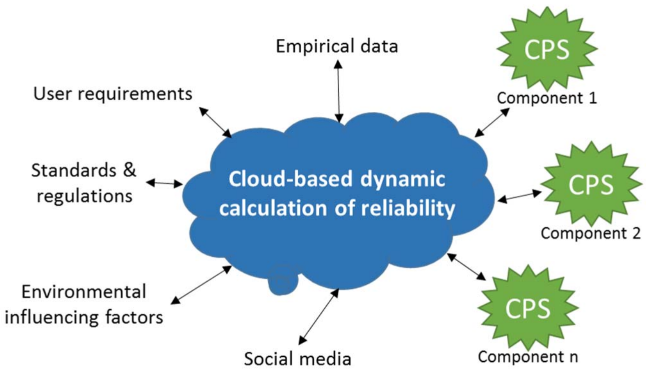

Figure 1: Basic structure of the system for a dynamic reliability calculation

As can be seen in Figure 1, a cloud-based application is used for dynamic calculation. The main focus is on interlinking different sources of information. Each component carries its basic failure rate, which is provided by the component manufacturer. The environmental parameters that affect its reliability, such as temperature, humidity, vibration, etc., are provided by interlinking and big data analysis. Furthermore, parameters such as standards, customer requirements, empirical data of the manufacturer, etc., are included in order to increase the accuracy of the overall reliability. Another aspect is the integration of social networks. This provides the possibility to include the experience of other users in the calculation. This allows to calculate the reliability dynamically, and therefore the availability of the system can be recalculated at runtime.

For the realisation of the scenario outlined in Figure 1 using agents, all sources of information are firstly encapsulated by an agent (see Figure 3). Each component agent represents one of the physical components, and each agent contains or calculates the reliability of its component locally. Due to the usually limited computing power of individual components, the local reliability calculation does not take dynamic influence parameters into account. The social media agent collects reliability-related information, such as evaluations, reviews or user complaints from search engines, newsgroups, social networks etc. This information can then be proactively passed on to corresponding component agents and the agent can react on inquiries from component agents for searching relevant information. Furthermore, the gathered information is made available for the reliability calculation in the cloud. The empirical data agent, the user requirements agent, the standards & regulations agent, and the environmental influencing factors agent all act in a similar manner at the same time. These agents are persistent and continuously collect information that is, or could be, relevant to the reliability calculation. All agents who represent a source of information can provide the information they have gathered to calculate the reliability dynamically in the cloud. This cloud calculation part is likewise agent-based and has been described in more detail in [17]. Before designing the necessary agent structure, the calculation of the total failure rates is briefly summarized (more details in [17]).

A. Failure rates

The failure rate of a cyber-physical system is the sum of three different partial failure rates, which are influenced by different factors which will be described in the following. The influencing factors, and therefore also the partial failure rates, are time-dependent and can change during the operating time of the component (see Figure 2). Here, only the relationship with the total failure rate is considered, which can be represented by the following formula (1). Here, n is the number of existing components. The failure rate consists of three elements which namely are (1) the predetermined manufacturer-related failure rate, (2) the human failure rate and (3) the influencing factors failure rate. A detailed description of each partial failure rate is given in the following paragraphs.

$$\lambda_i(t) = \lambda_{i,MF} + \lambda_{i,HF}(t) + \lambda_{i,IF}(t) \quad for\ i = 1, \dots, n \quad (1)$$

The addition of the three partial failure rates corresponds to a logical series structure of the three partial failure rates. This assumption can be made because the partial failure rates are causally and temporally independent from each other and therefore contribute to the overall failure rate of the considered component.

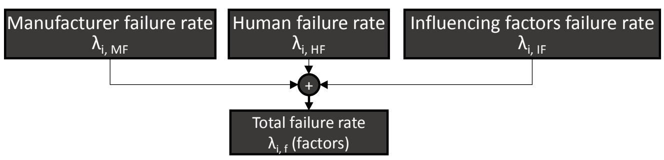

Figure 2: Calculation of the total failure rate from partial failure rates

Preconditioning: A predetermined, manufacturer-related failure rate λ_{MF} represents only an average value, which is usually obtained by testing several similar components. The actual failure rate of the component used can deviate from the predetermined one. All factors occurring here are known before startup and are time-independent. Each factor in this class changes the failure rate by a predetermined percentage p_j from the manufacturer-related failure rate $\lambda_{i,\mathrm{MF}}$. The factor p_j depends on the classification of each factor with respect to its influence quantity. The classes range from small to medium and large. The classification of each factor in one of the above categories is determined by the user.

Human reliability: Every interaction between human and machine includes a potential source of danger for people, the system and its environment. In the following, it is assumed that all interactions can be recorded and stored in a database. The database is located in a cloud and is aware of every component in the system. The Industry 4.0 concept makes such an assumption feasible.

Every interaction has the goal of performing a specific task, such as entering a target temperature in a control system. By detecting the interaction, we can also register if the task performed has led to an error or not. The HEP (**H**uman **E**rror **P**robability) can be calculated as a result. This gives the percentage of errors in each recorded task out of the total number of errors (see also Equation (2)).

$$HEP = \frac{Number\ of\ incorrect\ performed\ interacitons}{Number\ of\ all\ performed\ interactions} \qquad (2)$$

Influencing factors: The third and last category is formed by the influencing factors. All the time-dependent factors are in this category. Each of these factors has an allowable maximum, minimum and standard value. Furthermore, each factor is classified separately for each component. The division into classes is required to indicate the importance of the considered factor for each component. The greater the deviation from the standard value of a factor and the higher it is ranked, the greater the impact on the failure rate $\lambda_{i,IF}$ is [17].

B. Agent-based determination of partial default rates

The agent paradigm can be used to calculate the three previously introduced partial failure rates dynamically. Each of the partial failure rates is influenced by different parameters. The information for calculating the partial failure rates can be proactively gathered in a first step by the already introduced component agents or by the agents representing other information sources (social media agent, environmental influencing factors agent, etc.) and can then be used in the next step by the cloud-based dynamic reliability calculation. The agent structure of the dynamic reliability calculation system is shown in Figure 3. The agents represent the components and sources of information previously described in Figure 1.

Each of the partial failure rates of a component is also represented by an agent, who then communicates with the relevant respective component agents. If the requested or required information cannot be made available on the basis of the agents' current knowledge, the respective agent reacts by trying to obtain or record the desired information. For example, environmental parameters can be measured via suitable sensors. In case that the manufacturer ascertains a changed basic failure rate for a component through more detailed investigations, this can also be transmitted from the empirical data agent to the agents in the dynamic reliability calculation system. User interactions and their success rate can be determined, for instance, from analyzing log files or databases, as long as the components themselves log these user actions. It is also conceivable that information on possible problems with using a component can be obtained from social media channels.

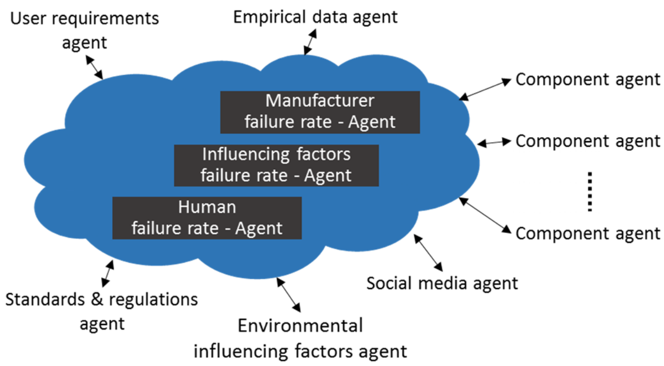

Figure 3: Basic structure of the system for dynamic reliability calculations

The three partial failure rate agents of a component calculate their respective value persistently and therefore make use of a further property of the agent paradigm. By adding the partial failure rates according to Equation (1), the failure rate $\lambda_i(t)$ of component i is permanently calculated during runtime. This can then be used to dynamically calculate the reliability of the overall system according to the underlying reliability block diagram.

V. Prototype to Visualise the Effects of Dynamic Influencing Factors

In order to better demonstrate and illustrate the influence of dynamic parameters on the reliability of a system, a cyber-physical system scenario has been developed, showing the production of adhesive tape in a fictional factory of the future. Different events happen during the lifetime of the factory, all of which have an impact on the reliability. Examples of these include: the failure and repair of a component, different ambient temperature and humidity to simulate seasons, errors in human-machine communication, etc. Then the course of the failure rate and reliability over time is considered to see how much influence these factors have on the reliability.

As part of this, an application was implemented in Matlab/Simulink that provides a dynamic visualization of the failure rate and reliability timing, also taking user-defined factors into account. An example can be seen in Figure 4. The application simulates cyclic and acyclic events and calculates the failure rate as well as the reliability for the defined scenario. The simulation period in the example is 7 years long. User interactions, including operating errors, can also be simulated in the application but were not included in the example scenario. This would allow to simulate and calculate effects of a changing human reliability during the simulation period.

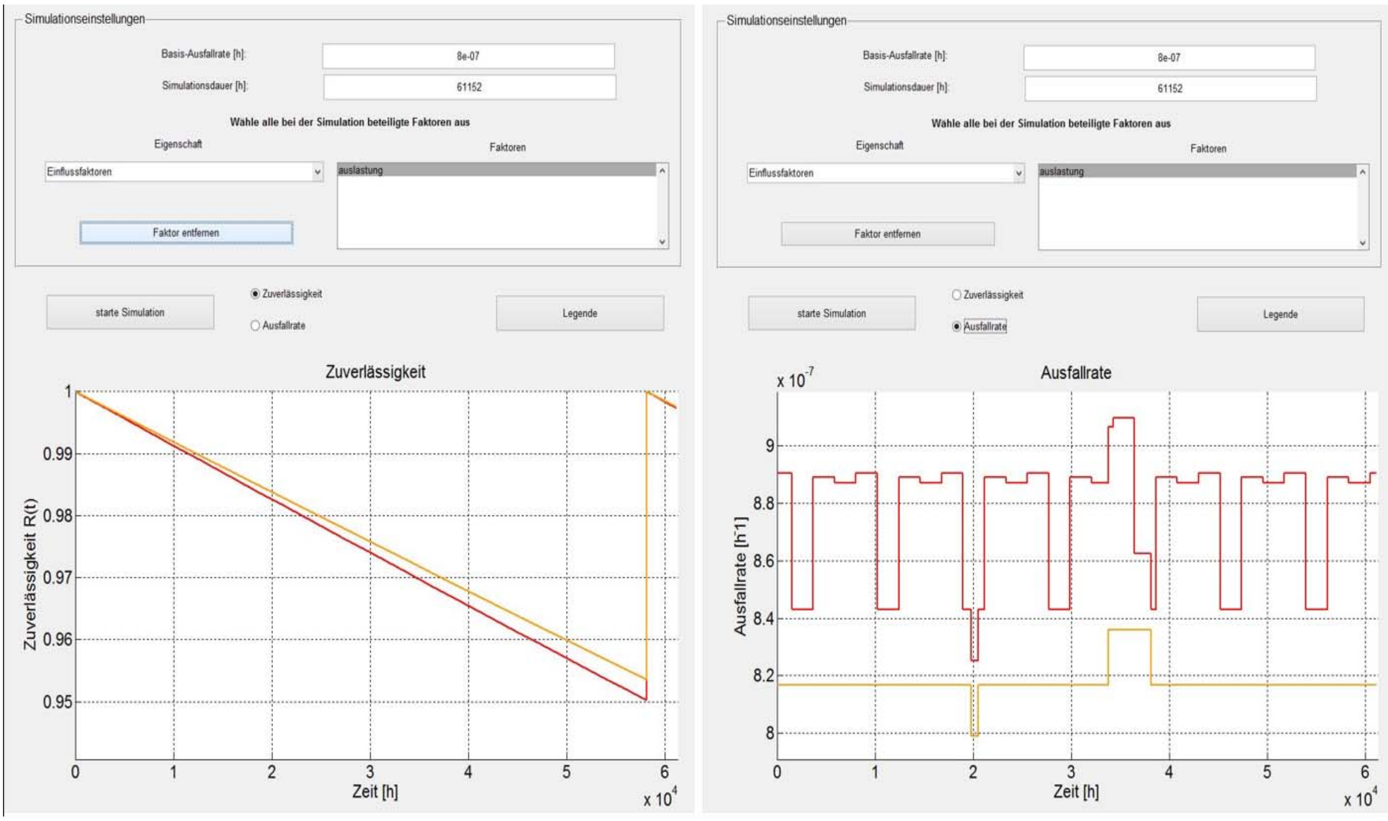

Figure 4: Illustration of the reliability with consideration of dynamic influencing factors (red) compared to the consideration without seasonal influences (orange)

The failure rate of a machine considered in the right part of Figure 4 is influenced by different ambient temperatures and humidity (red curve). The influence of the seasons can be identified by cyclical patterns in the failure rate. In addition, there is an acyclic effect at two points in time which also results in a change of the failure rate. One component fails after 118 weeks (t = 19824 h), which affects other components during the four-week repair period. These components must reduce their production volume which has a positive influence on the failure rate of the individual components and, therefore, also on the reliability of the entire system. A second failure occurs after 201 weeks (t = 33768h) when an adhesive mixer makes impermissible mixtures. The situation is detected by a sensor. However, the affected subsystem must be temporarily taken out of service. As there is a parallel, redundant mixer, the subsystem has to produce at a higher capacity. This change introduces additional changes to the system structure and therefore another reliability block diagram. The simulation of this modified block diagram results in a higher failure rate until the defective mixer is replaced.

Overall, however, despite a failure rate which is not constant over time, there is no large visible fluctuation in the reliability (Figure 4 - left), and the reliability follows a nearly linear course. It returns to value 1 as soon as the machine is completely repaired.

To detect the effect of the seasonal influences, the course of the failure rate and reliability is then simulated again, this time without the two factors of ambient temperature and humidity (orange curve). The failure rate is largely constant and only changes during periods with a changing load (component failures). This also has consequences for the reliability: a different course can be seen which differs approximately 0.4 percentage points from the value calculated with seasonal influences.

VI. Conclusions and Outlook

In determining the reliability of cyber-physical systems, new approaches and concepts need to be developed to meet the demands of new technologies. The traditional approach of calculating reliability is no longer sufficient, as the static reliability of the individual components is known, but on the other hand, it can be determined more dynamically by accessing data collected at runtime (enabled by the networking of components). In addition, the overall reliability of a system at the time of system development is not known because the composition of the components is likely to change at runtime.

In determining reliability, the Internet of Things and the "cloud" provide ways to calculate a system dynamically and to consider the influence of various factors at runtime. In this paper, parts of this data have been used to determine reliability more accurately. Different types of data have been included in the calculation, such as "empirical data" or the "human factor".

It was discussed how the reliability could be determined more accurately based on dynamic, time- and factor-dependent failure rates. Furthermore, the total failure rate was divided into three part failure rates, all of which are subject to different influences. Finally, a scenario was used to demonstrate the mode of action of these factors using an example.

The concept presented for the dynamic determination of reliability uses the agent-oriented way of thinking in order to develop an assistance system in the field of reliability calculation. It has been shown that concepts such as autonomy, persistence, interaction and proactivity can be used to create a dynamic reliability calculation system. Through the use of agent-specific properties, it is possible to include runtime information in the calculation processes which are not taken into account in conventional static reliability analysis. By including a large number of external, highly dynamic sources of information, such as social media, not only a-priori knowledge is available for the calculation of reliability, but also up-to-the-minute information. This approach does justice to current developments in smart components, and provides more options for reliability analysis than existing methods.

The concept should now be verified in the field with the help of empirical data in a further step. It can also be extended to include even more data from the cloud in the calculation. The role and possibilities of the social media agent also need to be further investigated. In particular, the content-related interpretation of the component reports poses a challenge.

Another challenge that arises in the dynamic reliability calculation is the automated detection of the runtime-changeable system structure, which has not been taken into consideration in this paper. Reconfiguration of the system during runtime or the entry or exit of system participants that were not known or intended at design time will result in frequent changes in the system structure. The interconnectedness of the individual components can therefore no longer be represented by a static reliability block diagram. The definition of system boundaries is no longer clear as soon as several systems become dependent on one another in terms of information technology and/or energy. It is therefore necessary to develop suitable methodologies and procedures that capture the system structure at runtime and therefore make the dynamics of the system structure available to the reliability calculation, in addition to the dynamics in the influence parameters on the reliability. The solution for this challenge will be addressed in future studies.

REFERENCES

[1] acatech – German National Academy of Science and Engineering e. V.: „Industrie 4.0 Maturity Index – Managing the Digital Transformation of Companies." Berlin, 25. April 2017.

[2] T. Bauernhansl, M. ten Hompel and B. Vogel-Heuser, „Industrie 4.0 in Produktion, Automatisierung und Logistik." Wiesbaden, Springer Vieweg, 2014.

[3] R. Gulati, J. Kahn, R.Baldwin, „The Professional's Guide to Maintenance and Reliability Terminology." Reliabilityweb.com Incorporated, 2010.

[4] N. Jazdi, „Lecture script: Zuverlässigkeit und Sicherheit von Automatisierungssystemen." IAS, University of Stuttgart, 2017.

[5] L. Urbas, „Lecture script: Prozessleittechnik 2-Zuverlässigkeitstechnik-Grundlagen." Fakultät Elektrotechnik & Informationstechnik Institut für Automatisierungstechnik, Professur für Prozessleittechnik, 2016.

[6] T. Beyer T, P. Goehner P, R. Yousefifar, KH Wehking, „Agent-based dimensioning to support the planning of Intra-Logistics systems." ETFA, IEEE 21st International Conference on 2016 Sep 6 (pp. 1-4).

[7] A. Wannagat, and B. Vogel-Heuser. „Agent oriented software-development for networked embedded systems with real time and dependability requirements in the domain of automation.", IFAC Proceedings Volumes 41, no. 2 (2008): 4144-4149.

[8] H. Mubarak, P. Goehner, „An agent-oriented approach for self-management of industrial automation systems." In Industrial Informatics (INDIN), 8th IEEE International Conference on (pp. 721-726), 2010.

[9] M. Rauscher and P. Goehner, „Agent-based consistency check in early mechatronic design phase." 19th International Conference on Engineering Design (ICED13), DVol. 9: Design Methods and Tools, Seoul, Korea, 19-22.08. 2013.

[10] T. Wagner, "Agentenunterstütztes Engineering von Automatisierungsanlagen" in IAS-Forschungsberichte Band 2008,1. Shaker Verlag, Aachen, 2008.

[11] T. Beyer, P. Göhner, N. Jazdi and R. Yousefifar, "Knowledge-based planning and adaptation of industrial automation systems". 2015 IEEE 20th Conference on Emerging Technologies & Factory Automation (ETFA), 2015.

[12] C. Dannegger, „Ready for Take-Off–Softwareagenten disponieren Flugcharter-Verkehr." In Agentensysteme in der Automatisierungstechnik (pp. 243-254). Springer Berlin Heidelberg, 2013.

[13] H. Kremer and C. Westerkamp, „Einsatz von Agentensystemen zur Optimierung der Logistik in Produktions-und Agrarprozessen." In Agentensysteme in der Automatisierungstechnik (pp. 187-205). Springer Berlin Heidelberg, 2013.

[14] S. Cavalieri, L. Fasanotti, S. Ierace, C. E. Pereira and M. Zuccolotto, "A biomimetic approach to distributed maintenance management based on a multi-agent system." Industrial Agents: Emerging Applications of Software Agents in Industry, 2015.

[15] P. Vrba, V. Mařík, P. Siano, P. Leitão, G. Zhabelova, V. Vyatkin and T. Strasser, "A review of agent and service-oriented concepts applied to intelligent energy systems." IEEE transactions on industrial informatics, 10(3), 1890-1903, 2014.

[16] VDI/VDE 2653-1: Agentensysteme in der Automatisierungstechnik, VDI/VDE-Richtlinien, VDI, Düsseldorf, 2010.

[17] N. Jazdi, "Dynamic calculation of the reliability of factory automation applications: Industry 4.0 applications," in 2016 IEEE International Conference on Automation, Quality and Testing, Robotics (AQTR), 2016.