DESIGN FOR RELIABILITY: AN EVENT- AND FUNCTION-BASED FRAMEWORK FOR FAILURE BEHAVIOR ANALYSIS IN THE CONCEPTUAL DESIGN OF COGNITIVE PRODUCTS

Thierry Sop Njindam and Kristin Paetzold
Universität der Bundeswehr München

ABSTRACT
Product complexity in modern engineering is rising at an ever-increasing rate for several reasons. On the one hand, designers are aimed at extending the functionality of products, thus, integrating them in human living environments and optimizing their interaction with humans. On the other hand, this functionality extension results from the synergetic integration of different disciplines. However, an important prerequisite for the market launch of these products is their ability to meet the previously defined requirements, particularly safety and reliability.
In this perspective, we proposed a framework for the early analysis of the functional behavior of cognitive products. We assume that the failure of a function is linked with a system internal state transition. It is then possible to model the sequencing of different possible states, and by this means different functional failures which lead to critical feared states, thus, taking into account the random nature of the occurring failures. The approach presented is explained using an extended stochastic petri net with switching time to model the failure behavior of a cognitive walker.

Keywords: Design for reliability, cognitive products, failure behavior analysis.

1 INTRODUCTION
From today´s perspective, the emphasis on increasing the functionality of modern engineering products is shaping the product development. Global competition, recent advances in technology, and increased customer expectations are some of the reasons for that [6].
In view of such a prospect, one of the challenges of innovation as center of modern technology product development is not only the extension of the functionality of classic mechatronic products, but also at the same time to secure this increased functionality over the whole product lifecycle. Addressing a successful design of these complex products with regards to their complexity and operational issues rely, therefore, on a validation of their attributes such as performance, functionality, at the very earliest stages of the product development. Furthermore, an essential ingredient for their successful market launch involves the ability of these products to meet the previously defined requirements and customer needs over a defined period of time [3]. This is defined as reliability and has been ranked by customers as one of the most significant purchase criteria [2].
New, innovative research projects and procedures have been recently elaborated according to the different forms of extending the functionality of mechatronic products. Cognitive products illustrate an example of those mechatronic products whose functionality goes beyond those of the classic ones and whose surplus value is formed by cognitive capabilities such as learning from events, acquiring knowledge, thinking of reasons, planning actions [5]. They consist of a physical carrier with embodied mechanics, electronics, microprocessor and software and are intended to strongly interact with users as well as among them [5]. According to that, their malfunction may have catastrophic impacts, compromising the existence of the product itself, users and environmental safety, hence, the need of efficient methods for an early failure behavior and performance analysis in the product development to avoid costly callbacks.
However, the starting point of the ensurement of the functionality of products at early stages in the conceptual design has been set up in recent works as the identification of potential occurring functional failures through abstraction and then the search for appropriate solution principles [7]. Moreover, the random nature of functional failures of these products needs to be brought to the

foreground. This will allow the designers on the one hand to ensure the safety despite random events and on the other hand to effectively and efficiently analyze the product´s performance, thus, including other issues as reliability, maintainability and so on.
The main contribution of this paper is that we suggest a framework to analyze the system´s behavior of cognitive products in presence of random functional failures during their operation. To cope with this issue, we make the assumption that highly variable conditions and unpredictability of the system´s environment do not affect its behavior. We restrict ourselves in this first approach to product internal states whereby functional failures are basically associated to state transitions within the system.
The paper is structured as follows: section 2 covers the issue of cognitive products and the basic functions they are supposed to fulfill. Moving to a more detailed analysis, we consider that product functions can be divided in sub-functions, whereby main functions should be, depending on the product requirements, prioritized over additional functions. Section 3 considers an early investigation of the failure behavior of cognitive products. In section 4, an extended stochastic Petri net will be used to illustrate the presented method applied during the conceptual design of a cognitive walker. The final sections 5 and 6 summarize the proposed approach and highlight proposals for future work.

2 COGNITIVE PRODUCTS AND FUNCTIONS

We mention cognitive capabilities within the product development for instance with the ability of a system to perceive its environment as well as its own system internal states, to match the insights gained with an available knowledge base, then in order to perform appropriate actions depending on specific circumstances. A technical system must have the following characteristics to be qualified as cognitive:

- perceive the environment as well as itself;
- encode input data;
- store information and retrieve them if necessary;
- think and solve problems;
- kinetic control abilities;
- use a suitable communication mechanism [20].

From this, we assume that a technical system or product is then considered as cognitive if it possesses all these six capabilities [20]. These capabilities can further be integrated in different variants and forms, thus, characterizing various levels of cognition.
A number of skills emerge from the cognitive capabilities a system is equipped with [21]:

- high degree of automation and from that the ability to act according to a specific situation;
- active interaction with its environment;
- ability to learn and to anticipate.

We emphasize the fact within the bounds of this consideration that such systems, being equipped with cognitive capabilities, exhibit emergence. This means they can develop a system behavior which does not unconditionally match with the one the developers thought ahead. Hence, the necessity to define constraints upon which the system will behave, according to the perceived information, internal knowledge, tasks to be performed. This can be suitably specified in the conceptual design where the design objectives as well as the corresponding product functions, based on the functional requirements, are defined in the functional space and refer (as shown in Fig. 1) to design parameters in the physical space [19].

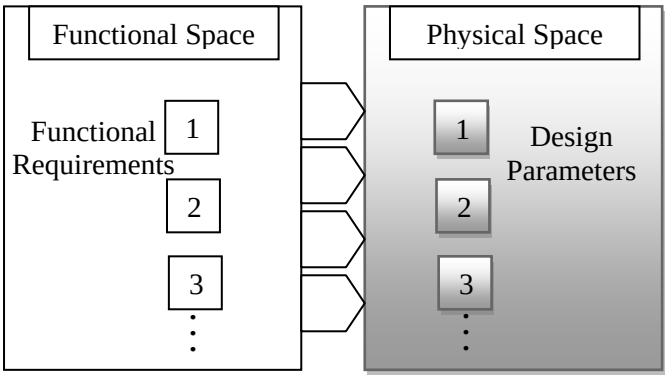

Fig.1: Mapping functional requirements to design parameters in the design process [19]

From the point of view of science and engineering, a function aims at representing physical or mathematical dependencies [18]. Pahl et. al. define in [1] this term as the "explicitly reproducible input / output relationship of a system" and refers to it at the same time as the abstract description of the transmission behavior of a system. Technical problems are then conceptually formulated on an abstract level using functions particularly at the very earliest stages of the product development process. This deserves the most attention since it is one of the critical stages in the design process [19]. Moreover, the system analysis is mostly supported by the functional description when developing new products in the field of mechanical engineering or multidisciplinary products such as cognitive products. The designers can, therefore, preserve the view of the whole by abstracting product-related complex issues. Starting from the definition the product´s overall task, functions can then be derived and, in turn can be hierarchically subdivided in sub-functions to find efficiently partial solutions in order to formulate the overall solution (figure 2).

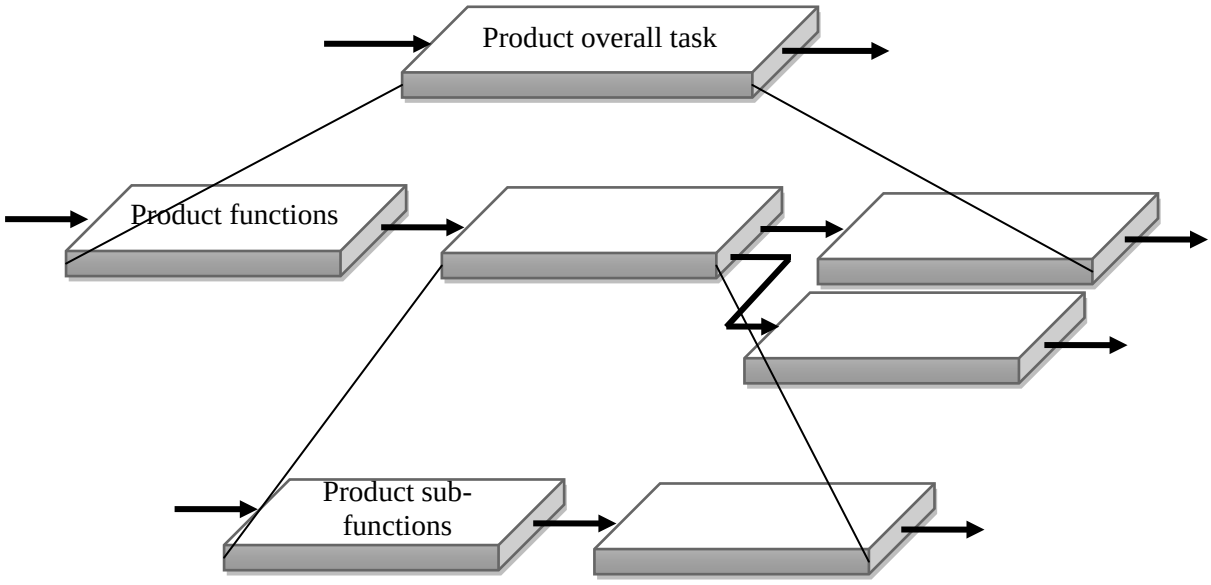

Fig.2: Hierarchical breakdown of product functions according to Pahl/Beitz [1]

3 RELATED WORK

Innovative products such as products equipped with cognitive capabilities, which are supposed to closely interact with humans in their living environments, are always associated with a high probability of the occurrence of failures and, therefore, are more prone to failures than less complex systems. Unreliability as well as damages to human beings, environment or equipment should be excluded right from the start of design and development. The only work addressing the safety of cognitive technical systems to the best of our knowledge was presented in [8]. Kain et. al. proposed in the paper a safety controller including safe sensors and safe actuators to meet safety requirements related to cognitive technical systems. This approach constituted the basis for the ensurement of safety of these technical systems equipped with cognitive mechanisms, in that the system simulation is performed for randomly chosen inputs to verify if a system critical state is attained. Nevertheless, this approach is neither suitable to investigate at early design stages the kind of failures to occur nor to take into account the associated performance degradation. In [9], Amalberti outlined the ultimate objective of the design of ultra-safe systems, namely the complete elimination of all technical breakdowns. This issue may refer within the design of cognitive products to the maximization of their operational safety on the basis of their inherent complexity and their non-deterministic behavior. This seems justifiable since cognitive products perform tasks by using various unpredictable procedures, thus, depending on such factors like the system knowledge, occurring events, user requests, boundary conditions.

We assume by considering these product properties that well known conventional end-to-end probabilistic analysis methods are not suitable for the failure behavior analysis of cognitive products

since they are subjected to random failures due to the harsh characteristic of the environment. Naresky defines them in [16] as “failures whose cause and / or mechanism make their time of occurrence unpredictable”. Furthermore, multiple statistically independent failure mechanisms varying from hardware wear out, software bugs and component aging, or their combination can be responsible for the system failure.
We also assume in order not to overload the scope of this work that product functions, forasmuch as no failure occurs, are fulfilled.
In our previous work, we qualified cognitive products as reparable systems by considering their capabilities [11]. They can either in some cases repair themselves or can be suggested to external repair. Moreover, we also assume that the system will be restored after the reparation as good as it was before the failure occurred. The troubleshooting does not cause additional failures. So we defined in that work a failure behavior model containing 5 states in which 4 stages of performance degradation coupled with the failures of functions and components were defined. We emphasize that vital cognitive functions such as perception, cognitive control and action play a primordial role. It is evident that these functions are necessary for the users’ safety and the basic product goals. The failure of these fundamental functions implies for example, that the product can neither be aware of dangerous situations nor can plan actions to avoid them. The system moves then to a safety critical state.
As sketched in the introduction, we aim at an efficient method for the analysis of the failure behavior of cognitive products at early development stages. Our approach (see figure 3) starts with the initially defined requirements. All the features the product should have including technical, safety are supposed to be defined and we assume they are expressed in terms of functions. These functions can be further subdivided in sub-functions for further specification. We assign the failure of a function to a state transition within the system. An initial state is, thus, defined to characterize a state in which all cognitive functions are available. Functional failures are then linked to stages of deterioration and by this means to internal state transition up to a safety critical state or to a total failure where products can no more fulfill the basic user´s needs, according to the functional requirements. As already mentioned, cognitive products can be considered as reparable devices, hence the need to consider random failures and repair rates based on our assumptions. From this point, we can then randomly vary these failures and repair rates of various functions with the objective of analyzing which functional failures or sequence of functional failures will lead quickly to the critical feared state. Iterations as usual within the product development process are necessary to adjust, if possible, the previously defined requirements. The defined failure and repair rate can serve as objective for the further component development and can be, therefore, analyzed throughout the process development cycle.

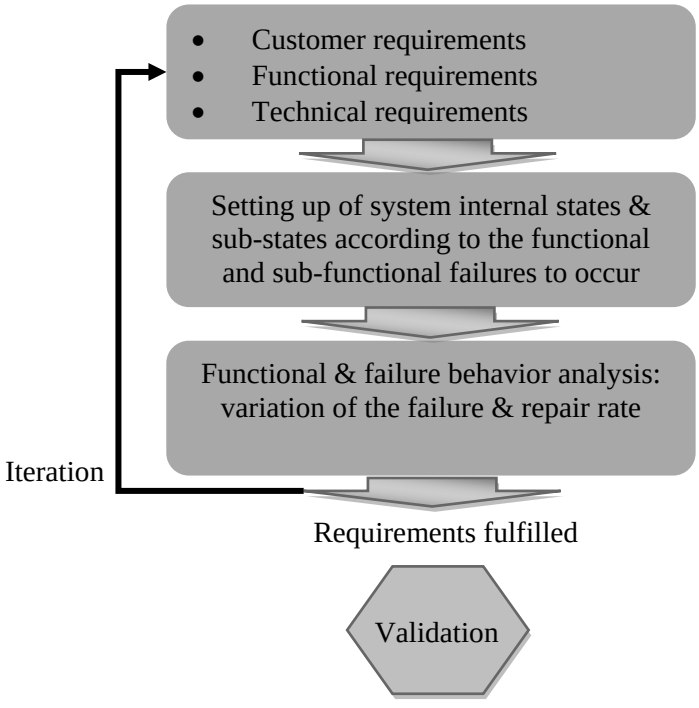

Fig.3: Framework for functional failure behavior of cognitive products

4 CASE STUDY: COGNITIVE WALKER

Let us consider the cognitive walker as an example of a product equipped with cognitive mechanisms, which aims at assisting elderly people in their daily duties. We build on the requirements expressed as functions and defined by designers for the development of this product. The product must fulfill the following functions:

- Environment Perception;
- Learning and Reasoning;
- Knowledge Processing;
- User Interaction;
- Cognitive Control;
- Action.

These main product features, represented during the conceptual design as functions can, furthermore, be subdivided in sub-functions, which are illustrated in the table 1 below.

Functions	Sub-functions
Environment perception	Recognizes an environment
	Identifies and locates obstacles
Learning & Reasoning	Trial-and-Error search for the best action
	Incorporates new information
	Adapts to users
Knowledge processing	Stores skills and information
	Internal representation of events and objects
	Improves its knowledge
User interaction	Adjusts to users
	Communicates with users (speech-based)
	Access a local area network
Cognitive control	Explains its actions
	Plans and generates appropriate actions
	Identifies risky situations
Action	Moves around an environment
	Avoids obstacles
	Performs scheduled tasks

Table 1: Required functions and sub-functions of the cognitive walker

We associate the failure of a specified function to a state transition as defined in table 2. And we start from an initial state in which all required functions are fulfilled. This state is referred to as Z_0. The system moves from this state Z_0 to a state Z_1 if the learning-and-reasoning function failed. This process goes so on until a safety critical state is reached. The transition to this critical feared state happens when a vital function such as perception, action as afore mentioned in the previous section has failed.

One novelty in our approach is the consideration of product sub-functions which according to table 1 can also be associated to sub-states. These sub-states linked with their corresponding sub-functions are illustrated for the state Z_0 in table 3. Quite to the contrary, the failure of a sub-function does not automatically imply a state transition. The system moves to the next state only if all the required sub-functions linked to a failed function fail.

The main objective of our approach is to analyze the random behavior of the failure of the product´s functions. We use an extended stochastic Petri net, which is shown in Fig. 3 to model the functional failure behavior of the walker. One essential characteristic of this kind of Petri net is the fact that each transition can be assigned to an arbitrary switching time [16].

Table 2: System internal states of the cognitive walker

State	Description
Z_0	Initial state; all functions are fulfilled
Z_1	Failure of the learning-and-reasoning function
Z_2	Supplementary failure of the knowledge function
Z_3	Additional failure of the interaction function
Z_4	Safety critical state; failure of either the perception function, action function, control function, or all required functions

Table 3: Sub-states of the State Z_1 according to the defined sub-functions

Sub-state	Description
Z_{11}	Failure of the sub-function: trial-and error search for the best action
Z_{12}	Failure of the sub-function: incorporates new information
Z_{13}	Failure of the sub-function: adapts to users

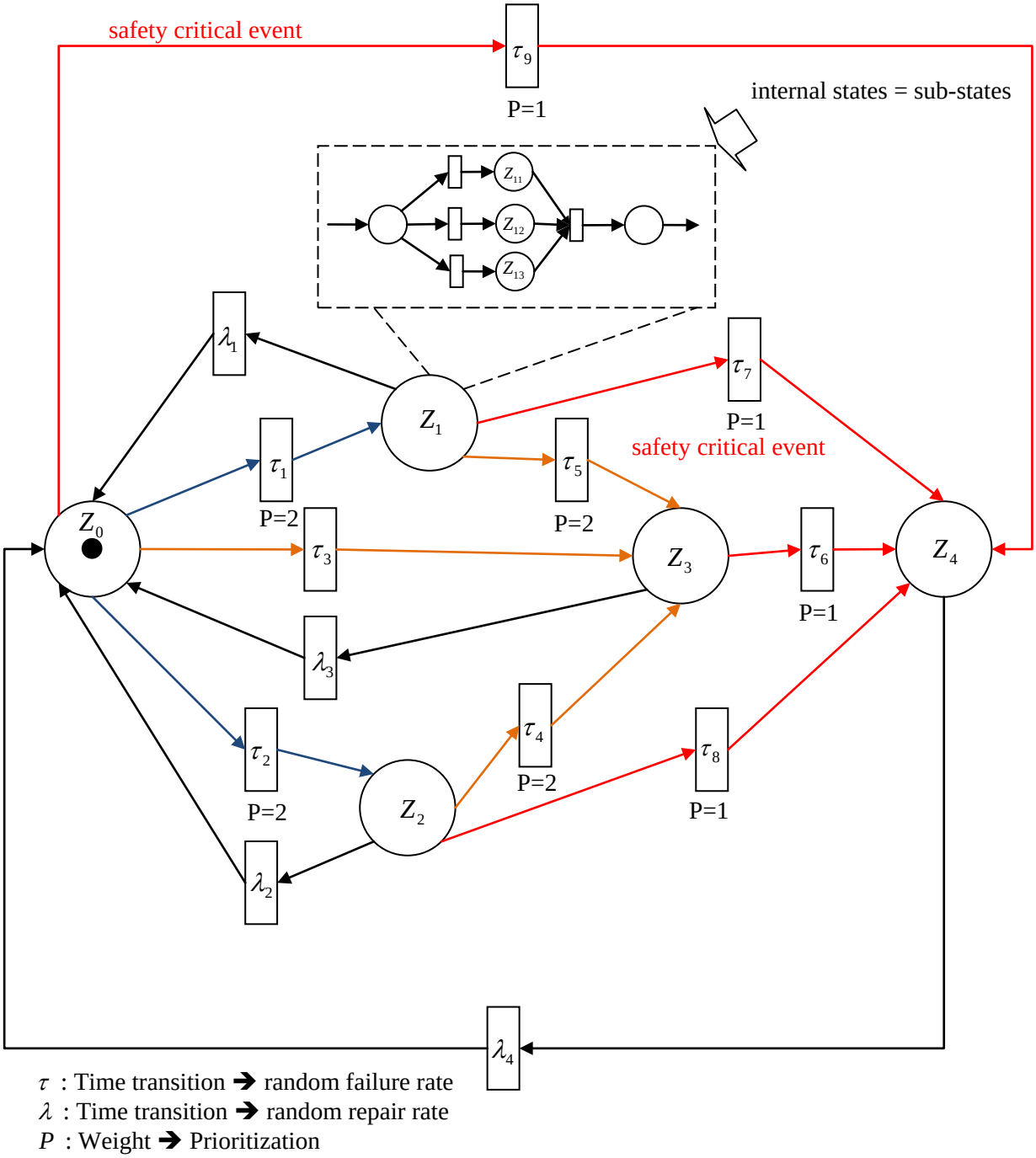

Fig.3: Stochastic Petri net with switching time for the failure behavior analysis of the cognitive walker

5 DISCUSSION

As sketched in the previous section 4, we aim at an efficient method for the analysis of the failure behavior of the cognitive walker. From reliability engineering, Markov models are suitable for the implementation of this endeavor. However, they quickly reach their limits beyond ten components [13]. The extended hierarchical stochastic Petri net model we have generated for the failure behavior analysis of the cognitive walker and shown in fig. 3 has the advantage of preventing which failure or sequence of failures leads the safety critical state. Furthermore, it can be used throughout the product development cycle.

Starting with the initial state Z_0 in which all predefined cognitive functions are available, the system moves either to the state Z_1 if the learning-and-reasoning failed, or to the state Z_2 if the knowledge-processing function failed, or to the state Z_3 if both of these functions are to be diagnosed as failed at the same time. Each state consists further of sub-states which characterize the predefined sub-functions. A prerequisite for the system to move from one state to another is the failure of all sub-functions of this state. τ_1 represents the random failure rate of the failure of the learning-and-reasoning function and λ_1 its random repair rate. The same also applies to the state Z_2 where τ_2 represents its random failure rate and λ_2 its random repair rate and so on. However, we assume that the random repair rate can only be considered if the walker can repair itself otherwise it will be submitted to external repair and moves then to the state Z_4.
As already mentioned, cognitive products are supposed to fulfill a set of basic functions during their operation. For this case, the perception of the environment, the cognitive control, the user interaction and action performing are vital for the walker to stay in operation otherwise it is considered as safety critical and then moves to the critical feared state Z_4. The concept of failure prioritization is of great interest here. What we mean by this is that all transitions within our model should be weighted so that the upkeep of these vital functions has the highest priority (P =1).
Finally, one weak point of this model is that a range for the random failures and repair rates is not specified in this application example. This is attributed to the fact that they are product and function specific. Empirical values could then be of used at this stage. Another possibility could be the variation of these values within a specified range. However, the non-fulfillment of the requirements engenders an iteration which can lead to an adjustment of the requirements such as the insertion of a redundancy to increase the safety. Should this not be the case, the validated random valued can be set as design parameters.

6 CONCLUSIONS AND FURTHER WORK

In this contribution, we proposed a framework to analyze and model the failure behavior and performance of cognitive products in the conceptual design phase. This approach is based on the functional requirements related to the product to be designed and combines the product functions with their event-based random failures.
In the second stage, system internal states and sub-states are defined according to the product´s functions and sub-functions. Starting from an initial state in which all product failures are available, a state transition refers to the failure of a function. The sub-states which are defined within states are indicative of the sub-functions' failures. Finally, the harsh interaction of cognitive products with their environment and with users implies random failures and repair rates, which according to the experience of the designers, can be varied to analyze which failure or combination of failures lead to critical feared states. This functional behavior analysis can then either be validated according to the defined requirements or be submitted to a further analysis if the requirements were not fulfilled.
Further work is needed in order to improve the proposed framework. For a simulation-based functional behavior analysis, a range for the switching time of the extended stochastic Petri net needs for example to be specified even if factors like failure and repair rates are product specific and depend on the designer's experience as well as on design parameters. The product´s main functions should also be weighted so that from this perspective their failures could be, if possible, avoided to the detriment of additional functions, which play a less important role.
However, we are aware that a physical embodiment including its simulation and validation is the ultimate last step of the product development. Furthermore, functional interactions as well as the impact of highly dynamic and harsh environmental conditions on the system behavior need to be considered. Anyway, based on the proposed framework we think that, thanks to a functional validation on this abstract level, the product development time could be reduced, the number of iterations and product recalls could be minimized.

ACKNOWLEDGEMENT

We gratefully acknowledge the close collaboration with Kristina Shea and Torsten Metzler from the Virtual Product Development Group of the Chair of Product Development of the Technical University Munich within the bounds of the development of cognitive products.

REFERENCES

[1] Pahl, G. et. Al, *Konstruktionslehre, 7th Edition*, 2002 (Springer Verlag)
[2] Birolini, A.: *Reliability Engineering: Theory and Practice, 6th Edition*, 2010 (Springer Verlag)
[3] Levin, M. and Kalal, T., *Improving Product Reliability*, 2003 (John Wiley & Sons)
[4] Nelson, W.: *Application of Functional Models to System Design, Operation and Performance Assessment*, In *IEEE International Conference on Systems, Man, and Cybernetics*, 1994
[5] Metzler, T. and Shea, K., *Cognitive Products: Definition and Framework, International Design Conference- Design 2010*, Dubrovnik, Croatia, May 2010
[6] Murthy, D.N.P et. Al.; *Investment in new product reliability*, In *Reliability Engineering and System Safety, Volume 94*, Issue 10, pp 1593 – 1600, October 2009
[7] Kurtoglu T. and Tumer I. Y.: *A Graph Based Fault Identification and Propagation Framework for Functional Design of Complex Systems, Journal of Mechanical Design*, ASME, 2008
[8] Kain et. Al., *Controller Architecture for Safe Cognitive Technical Systems*, In *26th Conference on Computer Safety, Reliability and Security*, Nuremberg, Germany, pp 518 – 531, 2007
[9] Amalberti, R.: *The paradoxes of almost totally safe transportation systems*, In *Safety Science*, Elsevier, 2001
[10] Beetz et al.: *Cognitive Technical Systems – What is the Role of Artificial Intelligence?* In *KI 2007: Advances in Artificial Intelligence, Vol. 4667, Berlin, Springer 2007, pp. 19-42.*
[11] Sop Njindam, T. and Paetzold, K.: *"Herausforderungen der Entwicklung zuverlässiger kognitiver Produkte", Symposium Design for X*, Buchholz in der Nordheide, 2010
[12] Girault, C. and Valk, R.: *Petri Nets for Systems Engineering – A Guide to Modeling, Verification, and Applications*, 2002 (Springer Verlag)
[13] Mihalache et. Al., *Reliability Analysis of Mechatronic Systems Using Censored Data and Petri Nets: Application on an Antilock Brake System (ABS)*, In *IEEE Transactions on Mechatronic Systems, 2006*
[14] Badenius, D., *Random Failures*, In *IEEE Transactions on Reliability*, 1970
[15] Naresky, J.J., *Reliability Definitions*, In *IEEE Standard Committee, IEEE Newsletter*, 1968
[16] Bause, F. and Kritzingen, P., *Stochastic Petri nets – an introduction to the theory*, 2002 (Vieweg Verlag, Braunschweig Wiesbaden, Germany)
[17] Guillerm, R.et. al: *Engineering dependability requirements for complex systems – A new information model definition, IEEE, 2010*
[18] Ehrlenspiel, K., *Integrierte Produktentwicklung – Denkabläufe, Methodeneisatz, Zusammenarbeit, 4. Überarbeitete Auflage*, 2009 (Hanser Verlag)
[19] Suh, N. P., *The Principles of Design*, Oxford Series on Advanced Manufacturing, 1990
[20] Paetzold, K., *Ethische Aspekte bei der Entwicklung kognitiver technischer Systeme für die Unterstützung bei demenziellen Erkrankungen*, In *Internation Conference on Engineering Design, ICED'07*, 28 – 31 august 2007, Cite des Sciences et de l´Industrie, paris, France
[21] Strube, G., *Modelling, Motivation and Action Control in Cognitive Systems*, In. *U. Schmid; J. Krems; F. Wysocki (Eds.), Mind Modelling*. Pabst, Berlin, 1998

Contact:
Dipl.-Ing. Thierry Sop Njindam, Prof. Dr. -Ing. Kristin Paetzold
Universität der Bundeswehr München / Institute of Technical Product Development
Faculty of Aerospace Engineering
Werner-Heisenberg-Weg 39
85577, Neubiberg
Germany
Tel: Int +49 089 6004 2821 or -2814
Fax: Int +49 089 6004 2815
E-mail: thierry.sop@unibw.de or kristin.paetzold@unibw.de
URL: http://www.unibw.de/lrt3

Dipl. -Ing. Thierry Sop Njindam works as a scientific assistant at the Institute of Technical Product Development of the Universität der Bundeswehr München and received his diploma thesis in mechatronics. His main research area focuses on the dependability of cognitive products, most notably on their reliability and safety from a systems engineering point of view.
Prof. Dr. -Ing. Kristin Paetzold studied mechanical engineering. She received her doctor´s degree in 2003 and worked afterwards as chief engineer at the Chair of Engineering Design in Erlangen. Since April 2009, she is a professor at the Universität der Bundeswehr München. Her research areas are cognitive technical systems, process- and workflow support, and senior-oriented product development.