Proceedings of the European Control Conference 2007
Kos, Greece, July 2-5, 2007

Behavior Based Definition of Dependability for Autonomous Mobile Systems

Jan Rüdiger, Achim Wagner and Essam Badreddin

Abstract— **In this paper a formal definition for dependability of autonomous mobile systems is proposed. The presented concept is based on the framework of Willems. The definition for dependability aims at providing the system designer/architect with a method for analysing the dependability of autonomous mobile systems.**

I. Introduction

Complex computing systems, such as network computers, computer controlled plants or flight controll systems need not only to fulfill their functional properties but also non-functional properties like availability, reliability, safety, performance, dependability etc. Non-functional properties reflect the overall quality of a system. Beside performance the dependability is getting a more important non-functional requirement of a system. According to [12] dependability is an integrated concept that further consists of the attributes

- availability,
- reliability,
- safety,
- integrity, and
- maintainability.

Depending on the research community and application of the system further attributes are allocated to dependability. Even if the attributes of dependability are known and accepted, a formal definition for dependability and some of its attributes is still missing. The lack of a formal definition makes the classification and comparison of dependable system nearly impossible.

The dependability of a system, however, is particularly important when dealing with autonomous or semi-autonomous systems. With an increasing degree of autonomy and with increasing safety requirements the requirements for dependability increase. Being able to measure and compare the dependability of these system is getting more and more important. For this purpose, a formal definition for dependability is needed not only for measuring and comparing the dependability of an existing system but also for developing techniques for designing new dependable systems.

The most common non-formal definition is given by Laprie in [14]. Based on this definition considerable research has been done by different groups about dependable systems. Two main approaches can be distinguished: The evaluation and validation approach tries to measure the dependability on an already built system; while the design approach tries to develop techniques for designing dependable hard and software systems.

In [1] a classification scheme for dependable systems is proposed based on availability, data integrity, disaster recovery, and security. The system is evaluated against a list of criteria for each dependability factor. The systems are then classified according to their application. This approach, however, lacks a formal method of how the systems are classified or how the attributes are measured and how they influence the classification. This makes, for example, the comparison of dissimilar architectures extremely problematic.

Benchmarking and fault-injection are another widely used technique to evaluate the dependability of complex computing systems ([2]–[7]). One advantage of the fault-injection approach is that it can take the human factor into account. Benchmarks, however, cannot predict the overall dependability of the system, but just measure the current dependability. The dependability in presence of a fault, for example, cannot be completely tested in advance with these methods.

Designing dependable software systems is e.g. covered in [8]–[10]. In [8] the *Software Architecture Model (SAM)*, a formal framework for specifying and analyzing software architecture, is extended to analyze non-functional properties like performance and dependability. In [9] a software pattern for building dependable software architecture is proposed. Even if the above design methods lead to a dependable system again a formal method is missing making it hard or impossible to verify or compare different architecture.

The definition for dependability proposed in this paper does not aim to be a general definition fitting any kind of system like computer systems in general, communication systems etc., but solely to autonomous mobile systems. The goal of this definition is to define dependability for a well known group of systems acting in a defined environment. The definition must be suitable measure and compare the dependability of existing autonomous systems and also to develop techniques for designing and developing new dependable systems. In addition the definition proposed should meet the common non-formal definition for dependability ([14]–[16]).

In this paper a definition of dependability in relation to the behavior of autonomous mobile systems is proposed. This definition is based on the framework of Willems ([11]). The elements from this framework used throughout this paper are introduced in Section II. In Section III the system and its boundaries for which the definition for dependability is proposed will be introduced and defined. Section IV outlines

ISBN 978-3-9524173-8-6

the main attributes of dependability and their non-formal definition which will then be formal defined for autonomous mobile systems. The final definition for dependability for autonomous mobile systems will be given and discussed in Section V.

II. Framework for a theory of dynamical systems

In the framework of Willems (see [11]) a system is defined in an universum $\mathbb{U}$. Elements of $\mathbb{U}$ are called outcomes of the system. A mathematical model of a system from a behavioral or black-box point of view claims that certain outcomes are possible, while others are not. The model thus defines a specific subset $\mathfrak{B} \subset \mathbb{U}$. This subset is called the *behavior* of the system.

A (deterministic) mathematical model of a system is then defined as:

Definition 2.1: A *mathematical model* is a pair $(\mathbb{U}, \mathfrak{B})$ with the universum $\mathbb{U}$ - its elements are called *outcomes* - and $\mathfrak{B}$ the behavior.

A. Dynamical System

A dynamical system is a set of trajectories describing the behavior of the system during the time instants of interest in $\mathbb{W}$.

In contrast to the state space representation, like $\dot{x} = f \circ x$, Willems (see [11]) defines a dynamical system as:

Definition 2.2: A *dynamical system* $\sum$ is a triple $\sum = (\mathbb{T}, \mathbb{W}, \mathfrak{B})$ with $\mathbb{T} \subseteq \mathbb{R}$ the time axis, $\mathbb{W}$ the signal space, and $\mathfrak{B} \subseteq \mathbb{W}^{\mathbb{T}}$ the behavior.

A dynamical system is described by a period of time $\mathbb{T}$, the signal space $\mathbb{W}$ and a set of time-trajectories $\mathfrak{B}$.

For a trajectory (an event) $w : \mathbb{T} \rightarrow \mathbb{W}$ it applies that:

- $w \in \mathfrak{B}$ the model allows the trajectory w
- $w \notin \mathfrak{B}$ the model forbids the trajectory w

The behavior $\mathfrak{B}$ is thus the set of all admissible trajectories. According to [11] a system can be further divided into sub-systems. Let $\Sigma_i = (\mathbb{T}, \mathbb{W}, \mathfrak{B}_i)$, then Σ_1 is called a *subsystem* of Σ_2 if $\mathfrak{B}_1 \subseteq \mathfrak{B}_2$.

B. Controllability

Further the controllability of a dynamical system is defined as a property of the system (see [11]):

Definition 2.3: Let $\Sigma = (\mathbb{T}, \mathbb{W}, \mathfrak{B})$, $\mathbb{T} = \mathbb{Z}$ or $\mathbb{R}$, be a time-invariant dynamical system. Σ is said to be controllable if for all $w_1, w_2 \in \mathfrak{B}$ there exists a $t \in \mathbb{T}$, $t \geq 0$, and $w : \mathbb{T} \cap [0, t] \rightarrow \mathbb{W}$ such that $w' \in \mathfrak{B}$, with $w' : \mathbb{T} \rightarrow \mathbb{W}$ defined by:

$$w'_{(t')} = \begin{cases} w_{1(t')} & \text{for } t' < 0 \\ w_{(t')} & \text{for } 0 \leq t' \leq t \\ w_{2(t'-t)} & \text{for } t' > t \end{cases}$$

The definition is illustrated in Fig. 1. In this definition the trajectory w_1 is the past trajectory of the system, w_2 the desired future trajectory and w the controlled trajectory. A controllable system is a system that can be controlled from any trajectory $w_1 \in \mathfrak{B}$ to any other trajectory $w_2 \in \mathfrak{B}$.

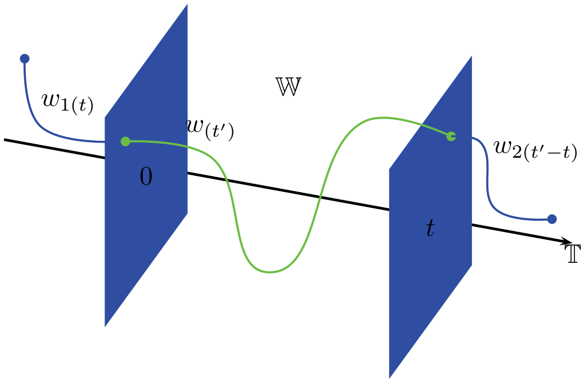

Fig. 1. Controllability: w_1 past trajectory of the system, w_2 desired future trajectory of the system and w the controlled trajectory

C. Autonomous System

In contrast to a controllable system an autonomous system is defined as a system which future is completely determined by its past.

Definition 2.4: Let $\Sigma = (\mathbb{T}, \mathbb{W}, \mathfrak{B})$, $\mathbb{T} = \mathbb{Z}$ or $\mathbb{R}$, be a time-invariant dynamical system. Σ is said to be *autonomous* if

$$\left\{ w_1, w_2 \in \mathfrak{B} \text{ and } w_{1(t)} = w_{2(t)} \text{ for } t < 0 \right\} \Rightarrow \{ w_1 = w_2 \}$$

The definition of controllable system and autonomous system form the the extreme points of possibilities. Most System will be a combination of these two. In [11] it is hence proposed that the behavior $\mathfrak{B}$ of a system can be divided into a controllable subsystem $\mathfrak{B}_c$ and into an autonomous subsystem $\mathfrak{B}_a$. For autonomous mobile systems the set $\mathfrak{B}_a$ depends on the view of the system. A system, for example, being capable of driving autonomously from one point to another without colliding will need different controllers like velocity controller or collision avoidance. Thus even when the behavior of the system appears autonomous to the user it is internally a set of controllers.

III. Behavior and Mission of an autonomous mobile system

Since its not the aim to define dependability in a global sense, the system and the environment must be defined. Each autonomous system is usually programmed to fulfill one or more tasks or missions. From a behavior point of view the autonomous system is given some behaviors which, when put together in the right way, will make the autonomous system able to fulfill its mission.

Definition 3.1: Let $\Sigma = (\mathbb{T}, \mathbb{W}, \mathfrak{B})$ be a time-invariant dynamical system then $B \subseteq \mathbb{W}^{\mathbb{T}}$ is called the set of *basic behaviors* $w_i(t) : \mathbb{T} \rightarrow \mathbb{W}$, $i = 1...n$ and $\mathbb{B}$ the set of fused behaviors.

B is a set of trajectories in the singal space $\mathbb{W}$. The set of basic behaviors B of an autonomous system, in contrast

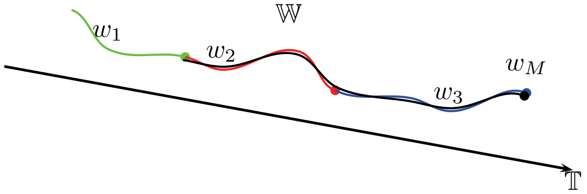

Fig. 2. A mission (black line) is accomplished by steering the system to the mission trajectory with the behavior w_1 and then steered along the mission trajectory with the behaviors w_2 and w_3

to the behaviors $\mathfrak{B}$ of a dynamical system as defined in Section II, is not the set of admissible behaviors, but solely those behaviors which are given to the system by the system engineer (programmer). Since the overall behavior $w(t)$ of an autonomous system is in many cases a fusion of one or more of the basic behaviors, the set $\mathbb{B}$ is defined which includes the basic behaviors together with behaviors generated by fusing the basic behaviors. Needless to say, that the set $\mathbb{B}$ strongly depends on the method used for fusing the basic behaviors.

For a given autonomous system it must not inevitably hold that $\mathbb{B} = \mathfrak{B}$. Usually the set $\mathbb{B}$ will be $\mathbb{B} \subset \mathfrak{B}$.

Definition 3.2: Let $\Sigma = (\mathbb{T}, \mathbb{W}, \mathfrak{B})$ be a time-invariant dynamical system. We say the *mission* w_m of this system is the map $w_m : \mathbb{T} \to \mathbb{W}$ with $w_m \in \mathfrak{B}$.

The mission, as defined here, is thus just a special trajectory or better a special behavior in $\mathfrak{B}$. The definition does not state that the mission trajectory must be $w_m \in \mathbb{B}$. The mission is defined to be element of the admissible trajectories $\mathfrak{B}$ of the system. Therefore we need to define when the mission w_m is accomplishable by a system with its given behavior set $\mathbb{B}$.

Definition 3.3: A mission $w_m \in \mathfrak{B}$ for a given dynamical system $\Sigma = (\mathbb{T}, \mathbb{W}, \mathfrak{B})$ with the behaviors $\mathbb{B}$ is said to be *accomplishable* by this system if for all $w_1 \in \mathfrak{B}$ there exists a $t \in \mathbb{T}$, $t \geq 0$, a behavior $w \in \mathbb{B}$, $w : \mathbb{T} \cap [0, t] \to \mathbb{W}$ and a behavior $w_2 \in \mathbb{B}$ such that $w' \in \mathfrak{B}$, with $w' : \mathbb{T} \to \mathbb{W}$ defined by:

$$w'_{(t')} = \begin{cases} w_{1(t')} & \text{for } t' < 0 \\ w_{(t')} & \text{for } 0 \leq t' \leq t \\ w_{2(t'-t)} & \text{for } t' > t \end{cases}$$

and

$$w'_{(t')} = w_m \quad \text{for } t' > t$$

The definition is illustrated in Fig. 2. It extends the definition of a controllable system. A mission w_m is said to be accomplishable if the system can be steered by a behavior $w \in \mathbb{B}$ from any past trajectory $w_1 \in \mathfrak{B}$ to the given mission trajectory $w_m \in \mathfrak{B}$ and is then able to follow the trajectory w_m. In Fig. 2 the black trajectory is the desired mission trajectory. The system is steered to the mission trajectory with the behavior w_1 and is then steered along the mission trajectory with the behaviors w_2 and w_3. The difference to the definition of controllability (see Section 2.3) is that the controll trajectory w and the future trajectory w_2 are not taken from the set $\mathfrak{B}$, but from the set of fused behaviors $\mathbb{B}$.

The above definition does not define any time constrains for accomplishing the mission nor does it define any mission quality factor up to which the mission is said to be accomplished. For this a partial mission accomplishment is defined.

Definition 3.4: A mission $w_M \in \mathbb{W}$ for a system $\Sigma = (\mathbb{T}, \mathbb{W}, \mathfrak{B})$ is said to be partially accomplished if the mission is accomplished according to a given performance index g_m.

This performance index can be for example a distance up to which a given position needs to be reached. The mission will then be seen as accomplished if the robots gets in a radius ε to the desired position.

IV. Attributes of Dependability

Before going further towards the definition for dependability for autonomous mobile systems first the basic attributes of dependability must be defined for systems as defined in the last section. According to [12] those attributes are:

- **Reliability** continuity of correct service,
- **Availability** readiness for correct service,
- **Safety** absence of catastrophic consequences for the user(s) and the environment,
- **Integrity** absence of improper system state alteration and
- **Maintanability** ability to undergo modifications and repairs.

To evaluate the dependability of a system one or more of these attributes is needed, depending on the the application. For a definition of dependability for autonomous mobile systems only the attributes reliability, availability, safety and maintainability are taken into account and thus defined here. The above definitions for the attributes of dependability are rather un-formal and thus not suitable for a mathematical analysis. For each of the attributes the non-formal definition is taken and a formal definition for autonomous mobile systems, as defined in Section III, is proposed.

A. Reliability

A common (see e.g. [16]) un-formal definition for reliability is:

> Reliability $R|_t$ is the probability that the system will operate correctly in a specified operating environment in the interval $[0, t]$, given that it worked at time 0.

An autonomous system is, thus, said to be reliable if the system state does not leave the set of admissible trajectories $\mathfrak{B}$. We can define the reliability of a system as:

Definition 4.1: Let $\Sigma = (\mathbb{T}, \mathbb{W}, \mathfrak{B})$, $\mathbb{T} = \mathbb{Z}$ or $\mathbb{R}$, be a time-invariant dynamical system. The system is said to be *reliable* in the period $[0, t]$ if for all $0 \leq t_1 \leq t$ the system state is $w(t_1) \in \mathfrak{B}$. Correspondingly, the *reliability* of the system is the probability that the system is reliable.

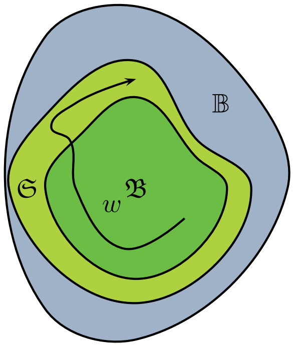

Fig. 3. Safety: The system trajectory w leaves the set of admissible trajectories $\mathfrak{B}$ but is still considered to be safe since it remains inside $\mathfrak{S}$

B. Availability

Availability is typically important for real-time systems where a short interrupt can be tolerated if the deadline is not missed.

> Availability $A|_t$ is the probability that a system is operational at the instant of time t.

In contrast to reliability the availability is defined at a time instant t while the reliability is defined in a time interval.

Definition 4.2: Let $\Sigma = (\mathbb{T}, \mathbb{W}, \mathfrak{B})$, $\mathbb{T} = \mathbb{Z}$ or $\mathbb{R}$, be a time-invariant dynamical system. The system is said to be *available* at time t if $w(t) \in \mathfrak{B}$. Correspondingly, the availability of the system is the probability that the system is available.

C. Safety

From a reliability point of view, all failures are equal. In case of safety, those failures are further divided into *fail-safe* and *fail-unsafe* ones. Safety is reliability with respect to failures that may cause catastrophic consequences. Therefore safety is unformaly defined as (see e.g. [16]):

> Safety $S(t)$ of a system is the probability that the system will either perform its function correctly or will discontinue its operation in a fail-safe manner.

For the formal definition of safety an area $\mathfrak{S}$ is introduced, like in [17]–[20], which leads to catastrophic consequences when left. In [17]–[20] it is, however, assumed that this *Dynamic Safety Margin* is fully contained in the stability region while $\mathfrak{S}$ is defined to be around $\mathfrak{B}$ here. This margin is, like $\mathfrak{B}$, highly system specific, but can be set equal to $\mathfrak{B}$ for a restrictive system.

Definition 4.3: Let $\Sigma = (\mathbb{T}, \mathbb{W}, \mathfrak{B})$, $\mathbb{T} = \mathbb{Z}$ or $\mathbb{R}$, be a time-invariant dynamical system with a safe area $\mathfrak{S} \supseteq \mathfrak{B}$. The system is said to be *safe* if for all $t \in \mathbb{T}$ the system state $w(t) \in \mathfrak{S}$.

This definition is consistent with the idea that a safe system is either operable or not operable but in a safe state.

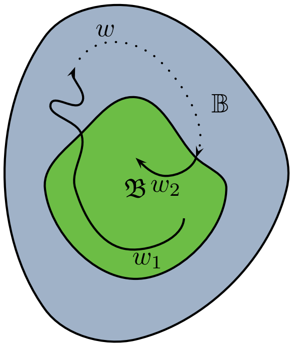

Fig. 4. Maintainability: The system trajectory w_1 leaves the set of admissible trajectories $\mathfrak{B}$ and is steered back to $\mathfrak{B}$ with the trajectory $w \in \mathbb{B}$

D. Maintainability

A maintainable system is „able to react“ either autonomously or by human interaction to changes in the system and the environment.

> Maintainability is the ability of a system to undergo modification and repairs.

A formal definition of maintainability for autonomous mobile robots is

Definition 4.4: A dynamical system $\Sigma = (\mathbb{T}, \mathbb{W}, \mathfrak{B})$ with the behaviors $\mathbb{B}$ is said to be *maintainable* if for all $w_1 \in \mathbb{W}$ a $w_2 \in \mathfrak{B}$ and a $w : \mathbb{T} \cap [0, t] \to \mathbb{W}$ exist, with $w' : \mathbb{T} \to \mathbb{W}$ defined by:

$$w'_{(t')} = \begin{cases} w_{1(t')} & \text{for } t' < 0 \\ w_{(t')} & \text{for } 0 \leq t' \leq t \\ w_{2(t'-t)} & \text{for } t' > t \end{cases}$$

The definition is illustrated in Fig. 4. Again the definition of a controllable system is extended. While in the definition of controllability the past trajectory must be in the set of allowed trajectories $\mathfrak{B}$ the past trajectory can now be in $\mathbb{W}$. For the system to be maintainable a behavior $w \in \mathbb{B}$ must exist that can steer the system from any state $w_1 \in \mathbb{W}^{\mathbb{T}}$ back to an allowed trajectory $w_2 \in \mathfrak{B}$. For this purpose the set $\mathbb{B}$ must include trajectories which can steer the system from every state $w \in \mathbb{W}^{\mathbb{T}}$ back to a state $w \in \mathfrak{B}$. This seems hardly accomplishable in practice but since the maintainability depends mainly on the set $\mathbb{B}$ a good behavior fusion algorithm for the basic behaviors B can increase the maintainability.

V. Dependability of Autonomous Mobile Systems

After having defined the system and its boundaries, a formal definition of dependability for autonomous mobile systems is proposed. For this the common non-formal definitions for dependability are used and their ideas transfered to the system defined in the previous sections.

> **Badreddin** [13]: Dependability in general is the capability of a system to successfully and safely fulfill its mission.
>
> **Carter** [15]: A system is dependable if it is trustworthy enough that reliance can be placed on the service it delivers.
>
> **Laprie** [14]: Dependability is that property of a computing system which allows reliance to be justifiably placed on the service it delivers.
>
> **Dubrova** [16]: Dependability is the ability of a system to deliver its intended level of service to its users.

All three definitions have in common that they define dependability on the service a system delivers and the trust that can be placed on that service. The service a system delivers is the behavior as it is perceived by the user, which in this case is the mission of the system.

Definition 5.1: A time-invariant dynamical system $\Sigma = (\mathbb{T}, \mathbb{W}, \mathfrak{B})$ with the behaviors $\mathbb{B}$ and a mission $w_m \in \mathfrak{B}$ is said to be (gradually) *dependable* in the period $T \in \mathbb{T}$ if, for all $t \in T$, the mission w_m can be (gradually) accomplished.

To accomplish a mission, as defined in section III, the state w of the system must be $w(t) \in \mathfrak{B}$. To be dependable this must apply for all $t \in T$. This assumption is equivalent to the *trust* mentioned in the non-formal definitions that is placed on the service the system delivers. To ensure $w \in \mathfrak{B}\ \forall t \in T$ the reliability, availability, safety and maintainability of the system must be designed such that the mission is accomplishable throughout the period T. The above definition is thus just a transformation of the common non-formal definitions for dependability in the framework defined in Section III. The advantage, however, is that with this definition, at least for a special group of systems, the dependability can be mathematical analyzed. This makes it possible to directly compare different strategies for improving the dependability of a system and to compare the dependability of different systems.

The set $\mathfrak{B}$ together with the set $\mathbb{B}$ are the key factors for the dependability of the overall system. While the set $\mathfrak{B}$ depends on the system and its environment and can only be influenced during the design, the set $\mathbb{B}$ is influenced by the amount of behaviors given to the system and the algorithm used for fusing them. A backup drive for a mobile robot, for example, will increases the dependability of the system via the reliability and availability since it increases the set $\mathfrak{B}$. An alternative position controller, using a slower but less hardware critical approach, will also increase the dependability of the system since it increases the set B and $\mathbb{B}$.

At runtime the dependability of the system again depends on the two sets $\mathfrak{B}$ and $\mathbb{B}$. A fault in the amplifier of the motors, for example, will decrease the set $\mathfrak{B}$ making it harder or even impossible to accomplish the given mission. An error in the position controller, for example, will lead to a smaller set $\mathbb{B}$ which again can result in a loss of dependability.

With this definition, the test whether a given system is dependable for its mission w_m can be done on the one hand with the classical benchmark approach, with the already mentioned disadvantages, to measure the dependability of the system under different pre-defined circumstances. The advantage, however, is that the test can also be done during the operation of the system, thus having an online test for the dependability of the system. If a fault happens to the systems that could lead to the fact that the mission cannot be accomplished, different strategies can be tested online which ensure the dependability of the system.

Fortunately, the test for dependability of a system must not be done for the whole system at once or for the whole system at all. If the system Σ can be divided into smaller sub-systems $\Sigma_i\ i = 1..n$ all having their own sub-mission w_{mi} and if the algorithm for fusing those sub-systems preserves the dependability then the dependability of the whole system can be determined by measuring the dependability of the sub-systems.

Typically not all sub-systems are relevant for accomplishing the mission w_m the effort for measuring the dependability of the system can be furthermore reduced by partitioning the sub-systems Σ_i into those sub-systems which are relevant for the dependability Σ_{di} and those which are not.

Furthermore the proposed definition for dependability can not only be used to measure the dependability of a given system, but can also be used for the design and development of new dependable systems. New techniques for improving the dependability or designing new dependable systems can be easily compared.

Finally, this definition is not implementation dependend and can thus serve as a benchmark to compare different implementations of dependable systems.

VI. CONCLUSIONS AND FUTURE WORKS

A. Conclusions

Dependability is part of the non-functional properties of a system which reflect the overall quality of a system. Even if a lot of research is being done focusing on measuring and improving the dependability of systems a formal definition of dependability is still missing.

In this paper a formal definition for dependability of autonomous mobile systems is proposed. This definition is based on the framework defined in [11] where a system is defined solely by its behavior. The framework was extended by defining an autonomous mobile system as a dynamical system with a set of behavior B together with the set of fused behaviors $\mathbb{B}$ and a mission w_m. For these systems a formal definition for each attribute of dependability, as defined in [12], was proposed derived from the non-formal definition. Finally, a formal over-all definition for dependability of autonomous mobile systems

acting in a known environment was proposed.
With this definition different approaches for improving the dependability can be directly compared to each other.
Even though the definition is only related to autonomous mobile systems, the definition is compatible with the commonly used non-formal definition for dependability ([14]–[16]).

The key elements of the dependability as defined here are the sets B, $\mathbb{B}$ and $\mathfrak{B}$. To improve the dependability of a system those need to be further analyzed and optimized. Improving the dependability of a system can be done either by improving the system itself and thus the set $\mathfrak{B}$ or by improving the behaviors of the systems together with the algorithm for fusing them and thus the set B and $\mathbb{B}$.

REFERENCES

[1] Don Wilson and Brendan Murphy and Lisa Spainhower, Progress on Defining Standardized Classes for Comparing the Dependability of Computer Systems, DSN Workshop on Dependability Benchmarking, June 25, 2002.

[2] Karama Kanoun, Henrique Madeira, Jean Arlat, A Framework for Dependability Benchmarking, DSN Workshop on Dependability Benchmarking, June 25, 2002.

[3] Aaron B. Brown, Leonard C. Chung, and David A. Patterson, Including the Human Factor in Dependability Benchmarks, DSN Workshop on Dependability Benchmarking, June 25, 2002.

[4] Michel Cukier and Carol S. Smidts, Using Bayesian Theory for Estimating Dependability Benchmark Measures, DSN Workshop on Dependability Benchmarking, June 25, 2002.

[5] Ioana Rus, Victor Basili, Marvin Zelkowitz, Barry Boehm, Empirical Evaluation of Techniques and Methods Used for Achieving and Assessing Software High Dependability, DSN Workshop on Dependability Benchmarking, June 25, 2002.

[6] Arup Mukherjee, Daniel P. Siewiorek, Measuring Software Dependability by Robustness Benchmarking, IEEE Transactions on Software Engineering, vol. 23, no. 6, pp. 366–378, 1997

[7] J. Arlat, M. Aguera, L. Amat, Y. Crouzet, J.-C. Fabre, J.-C. Laprie, E. Martins, D. Powell, Fault Injection for Dependability Validation: A Methodology and Some Applications, IEEE Transactions on Software Engineering, vol. 16, no. 2, pp. 166-182, Feb., 1990.

[8] Tianjun Shi, Xudong He, Dependability Analysis using SAM, Proc. of the ICSE Workshop on Software Architectures for Dependable Systems, pages 37-42. June 2003.

[9] Lihua Xu, Hadar Ziv, Debra Richardson, Thomas A. Alspaugh, An Architectural Pattern for Non-functional Dependability Requirements, WADS '05: Proceedings of the 2005 workshop on Architecting dependable system,. St. Louis, Missouri, New York, USA. 2005

[10] M. Tichy, H. Giese, An Architecture for Configurable Dependability of Application Services, Proc. of the ICSE 2003 Workshop on Software Architectures for Dependable Systems. Portland, OR. April 2003. pp. 65-70.

[11] Jan C. Willems, Paradigms and Puzzles in the Theory of Dynamical Systems, IEEE Transactions on Automatic Control, Vol. 36 No. 3 March 1991.

[12] Algirdas Avizienis, Jean-Claude Laprie, Brain Randell, and Calr Landwehr, Basic Concepts and Taxonomy of Dependable and Secure Computing, IEEE Transactions on Dependable and Secure Computing, Vol. 1, No. 1, January-March 2004

[13] Lecture Notes, Safety and Dependability of Mechatronic Systems, ETH Zürich, 2002.

[14] J. C. Laprie, Dependable computing: BASIC concepts and terminology: in english, french, german, italian and japanese. Ed. Springer – Verlag, 1992

[15] W.C. Carter, A Time for Reflection, Proc. 12th Int. Symp. on Fault Tolerant Computing (FTCS-12) IEEE Computer Society Press Santa Monica, CA, USA, June 1982

[16] Elena Dubrova. Fault Tolerant Design: An Introduction.

[17] E. Badreddin, M. Abdel-Geliel. Dynamic Safety Margin Principle and Application in Control of Safety Critical System. IEEE International conference on control application (CCA 2004), September 2-4, 2004, Taiwan. 689-695.

[18] M. Abdel-Geliel, E. Badreddin. Dynamic Safety Margin in Fault Diagnosis and Isolation. European Safety and Reliability (ESREL) conf., Tri city Poland, June 27-30, 2005.

[19] M. Abdel-Geliel, E. Badreddin, A. Gambier. Dynamic Safety Margin in Fault-Tolerant Predictive Controller. IEEE Conference on Control Applications Toronto, Canada, August 28-31, 2005

[20] M. Abdel-Geliel, E. Badreddin, A. Gambier. Application of Model Predictive Control for Fault Tolerant System Using Dynamic Safety Margin. American Control Conference Minneapolis, Minnesota, USA, June 14-16, 2006