

Beiträge aus der Elektrotechnik

Kai Ding

**Dependability-oriented Design and Analysis of
Control Systems at the Model Level under Random
Hardware Faults**

 VOGT

Dresden 2019

Bibliografische Information der Deutschen Nationalbibliothek
Die Deutsche Nationalbibliothek verzeichnet diese Publikation in der
Deutschen Nationalbibliografie; detaillierte bibliografische Daten sind im
Internet über <http://dnb.dnb.de> abrufbar.

Bibliographic Information published by the Deutsche Nationalbibliothek
The Deutsche Nationalbibliothek lists this publication in the Deutsche
Nationalbibliografie; detailed bibliographic data are available on the
Internet at <http://dnb.dnb.de>.

Zugl.: Dresden, Techn. Univ., Diss., 2019

Die vorliegende Arbeit stimmt mit dem Original der Dissertation
„Dependability-oriented Design and Analysis of Control Systems at the
Model Level under Random Hardware Faults“ von Kai Ding überein.

© Jörg Vogt Verlag 2019
Alle Rechte vorbehalten. All rights reserved.

Gesetzt vom Autor

ISBN 978-3-95947-040-7

Jörg Vogt Verlag
Niederwaldstr. 36
01277 Dresden
Germany

Phone: +49-(0)351-31403921
Telefax: +49-(0)351-31403918
e-mail: info@vogtverlag.de
Internet : www.vogtverlag.de



**Zuverlässigkeitsorientierter Entwurf und
Analyse von Steuerungssystemen auf
Modellebene unter zufälligen
Hardwarefehlern**

Dependability-oriented Design and Analysis of Control
Systems at the Model Level under Random Hardware
Faults

Kai Ding

von der Fakultät Elektrotechnik und Informationstechnik
der Technischen Universität Dresden

zur Erlangung des akademischen Grades eines

Doktoringenieurs

(Dr.-Ing.)

genehmigte Dissertation

— * —

Vorsitzender: Prof. Dr.-Ing. habil. Dipl.-Math. Klaus Röbenack

Gutachter: Prof. Dr. techn. Klaus Janschek
Prof. Antoine Rauzy

Tag der Einreichung: 24.06.2019

Tag der Verteidigung: 07.10.2019

Acknowledgements

I would like to express my deepest appreciation to my advisor, *Professor Dr. techn. Klaus Janschek*, whose guidance helped me throughout the entire period of my Ph.D. research. I could not have imagined having a better advisor for my Ph.D. study. Without his advice and persistent support, this dissertation would not have been possible.

My sincere gratitude also applies to *Dr.-Ing. Andrey Morozov* for his considerable guidance, helpful advice, valuable information, steady encouragement, and constant support during my Ph.D. research. Without his assistance, it would not have been possible to conduct this research.

I would also like to acknowledge *Professor Antoine Rauzy* from the Department of *Mechanical and Industrial Engineering of Norwegian University of Science and Technology* for accepting the role of co-referee of my doctorate thesis as well as his overall interest in my work.

Additional thanks extend to all of my colleagues at the *Automation Institute at Technische Universität Dresden*, especially the group of *Model-based System Analysis*, for providing a fruitful working atmosphere.

Finally, I would like to recognize the German Research Foundation (DFG) under project No. JA 1559/5-1 for granting me the Ph.D. research.

Abstract

Model-based design is a common methodology in the development of embedded complex control systems. Control system engineers typically prefer to use MATLAB[®] Simulink[®] and suitable automatic code generators for the development and deployment of software. Embedded systems are subject to random hardware faults; bit-flips, for example, may affect random access memory (RAM) cells and central processing unit (CPU) registers and cause data errors that may propagate to critical system outputs and result in system failures.

From a dependability perspective, the design space of control systems includes the selection of a suitable (reliable) implementation of a control algorithm. Such algorithm can be implemented with model-based software development frameworks, such as Simulink using different, but functionally equivalent implementations. However, these functional equivalents may exhibit completely different reliability properties. This thesis proposes an analytical method for the evaluation of the reliability properties of control systems that are designed with Simulink models. The method is based on a transformation of the assembly code, which is generated from the Simulink model, into a formal stochastic error propagation model as well as its quantification through underlying Markov chain models and state-of-the-art probabilistic model-checking techniques. The application of the method to the functionally equivalent implementations can determine which one is less vulnerable to data errors due to random hardware faults.

Fault tolerance is significant to dependable system design. Control systems can be protected with fault tolerance mechanisms to increase the reliability. Redundancy is the key underlying concept for achieving fault tolerance that is usually implemented at the hardware or software level. In the case of model-based development, redundancy mechanisms are preferable for direct application at the model level (Simulink model level). This thesis introduces a systematic classification of fault-tolerant design patterns. Such patterns can be applied to the Simulink model to tolerate random hardware faults, and taken into account during the control system design. In addition, it is more transparent and convenient for control system engineers to directly protect vulnerable parts with fault tolerance mechanisms at the model level.

The rigorous reliability assessment of the embedded control systems must be conducted at the assembly level based on the modeling of data errors that occurred in RAM and CPU. However, the scalability of the assembly-level assessment method is challenging and even problematic in view of the state space explosion (SSE) problem of the underlying Markov chain models. The computational complexity may increase exponentially as the assembly code size increases. Moreover, the transformation from the Simulink models to the assembly code is a complicated procedure. It is also more convenient for control engineers to already be able to estimate reliability properties and implement possible reliability improvements at the model level in the early design phase, when the model-based design is actually applied. Therefore, this thesis proposes a model-level reliability evaluation of Simulink models to address the aforementioned problems. The efficiency of the proposed model-level evaluation is verified by a comparison of the reliability properties that are assessed at the assembly and model levels.

Contents

Abstract	v
Contents	vii
1 Introduction	1
1.1 Research objectives and contributions	2
1.2 Thesis structure	4
1.3 Bibliographic notes	5
2 Preliminaries	7
2.1 Dependability	7
2.2 Hardware faults, bit-flips, and soft errors	9
2.3 Model-based design of control systems	10
2.4 Dual-graph error propagation model	11
3 Reliability evaluation of control algorithm implementations at the assembly level	15
3.1 State of the art	16
3.1.1 DTMC-based modeling methods	16
3.1.2 Comparison with the AVF and PVF	17
3.1.3 Hardware fault injection at different levels	17
3.1.4 Implementation variants of a PID controller	19
3.1.5 Contribution of this method	20
3.2 Method overview	20
3.3 First implementation: separate blocks for the P , I , and D terms	21
3.3.1 Simulink model	21
3.3.2 Parameters of the PID controller	22
3.3.3 Generated C code	22
3.3.4 Compiled assembly code	24
3.3.5 Transformation of the assembly code into the DEPM	25
3.4 Reliability evaluation with the probabilistic modeling of data errors in RAM and CPU	28
3.4.1 Data errors in RAM	29
3.4.1.1 Data errors in a single variable	30

3.4.1.2	Data errors in all variables	31
3.4.2	Data errors in CPU	31
3.5	Applications of the analytical method to three other imple- mentations of the PID controller	32
3.5.1	Second implementation: a Discrete PID Controller block	32
3.5.2	Third implementation: a Discrete Transfer Function block	34
3.5.3	Fourth implementation: a Discrete State-Space block	35
3.6	Comparison of the reliability properties of the four PID con- troller implementations	36
3.6.1	Numerical results evaluated with fixed probabilities .	37
3.6.2	Numerical results evaluated with varying probabilities	38
4	Fault-tolerant design patterns	43
4.1	State of the art	43
4.1.1	Traditional hardware and software redundancy	43
4.1.2	Contributions of the new classification of fault-tolerant design patterns	44
4.2	Basic design patterns	46
4.2.1	Comparison pattern	46
4.2.2	Voting pattern	46
4.2.3	Sparing pattern	48
4.3	Combined design patterns	49
4.3.1	Comparison then sparing pattern	49
4.3.2	Comparison then voting pattern	49
4.3.3	Voting then comparison pattern	51
4.3.4	Voting, comparison then sparing pattern	51
4.4	Comparison of reliability properties of defined design patterns	51
4.5	Common implementations of defined design patterns	53
5	MORE: MOdel-based REDundancy for Simulink models	63
5.1	State of the art	64
5.1.1	Fault tolerance and redundancy	64
5.1.2	Soft error protections at the hardware, software, and model levels	64
5.1.3	Contributions of the MORE approach	65
5.2	Model-based redundancy approach	66
5.2.1	Implementation of the <i>voting pattern</i> in Simulink . .	66
5.2.2	Implementation of the <i>comparison pattern</i> in Simulink	67

5.2.3	Implementation of the <i>sparing pattern</i> in Simulink . . .	68
5.2.4	Implementation of the <i>comparison then sparing pattern</i> in Simulink	68
5.3	Reliability evaluations of Simulink models protected by MORE approaches	70
6	Model-level assessment of Simulink models	77
6.1	State of the art	78
6.1.1	Effectiveness of fault injection at the model level . . .	78
6.1.2	Numerical reliability evaluation at the assembly level vs. at the model level	79
6.1.3	Contributions of the model-level assessment	80
6.2	Reliability assessment at the model level	81
6.2.1	Reliability evaluation of individual Simulink blocks func- tions at the assembly level	82
6.2.2	Generated model-level DEPM from a Simulink model	84
6.2.3	Probabilistic modeling of data errors at the model level	85
6.3	Comparison of the model-level assessment and assembly-level assessment	87
7	Conclusion	91
7.1	Summary	91
7.2	Future work	93
Appendix A	PRISM models	97
References		101

List of Figures

1.1	Three key contributions of the thesis.	3
2.1	The dependability tree; the key focus of this thesis is i) the reliability evaluation of embedded control systems under random hardware faults developed using Simulink models and ii) the fault tolerance techniques that can be applied at the model level.	8
2.2	The fault-error-failure chain.	8
2.3	A reference example of a dual-graph error propagation model of a repairable power system.	12
3.1	An overview of the comparison of reliability properties of different implementations of a PID controller in Simulink.	15
3.2	A PID controller designed using separate blocks for the P , I , and D terms.	22
3.4	(a) The automatically generated model step function of the first Simulink PID controller in Figure 3.2 and (b) the compiled assembly code from the automatically generated step function of (a).	24
3.5	The DEPM that is generated from the compiled assembly code (Figure 3.4b) of the first implementation; the element FI models data errors in RAM, and the instruction elements model data errors in the CPU destination registers.	27
3.6	The modeling concept of data errors in RAM and CPU during the execution of the embedded system.	29
3.8	The step function of the second implementation.	33
3.9	The step function of the third implementation.	35
3.10	The step function of the fourth implementation.	36
3.11	The evaluated N_{err} (left y-axis) and P_{err} (right y-axis) of the PID controller with the modeling of data errors in RAM with varying data error probabilities p_{RAM}	39
3.12	The evaluated N_{err} (left y-axis) and P_{err} (right y-axis) of the PID controller with the modeling of data errors in CPU with varying data error probabilities p_{CPU}	39

4.1	The proposed classification of fault-tolerant design patterns; three basic design patterns (top) can form various combinations, and the four combined design patterns (bottom) are presented.	45
4.4	Comparison of the reliabilities of the introduced design patterns.	52
4.8	Hardware implementation of the <i>comparison then sparing pattern</i> : Pair-and-a-spare with error detection units.	60
4.9	Hardware implementation of the <i>voting, comparison, then sparing pattern</i> : NMR with spares.	61
5.1	An overview of the applications of fault-tolerant design patterns to a PID controller and the evaluation process.	63
5.2	Implementation of the <i>voting pattern</i> to the <i>P</i> term of a PID controller.	67
5.3	Implementation of the <i>comparison pattern</i> to the <i>P</i> term of a PID controller.	67
5.4	Implementation of the <i>sparing pattern</i> to the <i>P</i> term of a PID controller.	69
5.5	Implementation of the <i>comparison then sparing pattern</i> to the <i>P</i> term of a PID controller.	69
5.8	The DEPM that is generated from the compiled assembly code (Figure 5.7b).	73
6.1	An overview of the reliability evaluation of a Simulink model i) at the assembly level (top) and ii) at the model level (bottom).	78
6.2	The generated DEPM from the compiled assembly code of the output function of the Integrator block.	83
6.3	The database stores (a) the reliability properties of Simulink blocks functions, evaluated at the assembly level by a probabilistic modeling of data errors in RAM and (b) probabilities of an incorrect value in the output variable of Simulink blocks functions, evaluated at the assembly level by a probabilistic modeling of data errors in CPU ($p_{CPU} = 1e - 4$).	84

List of Tables

3.1	The data error activation probabilities in RAM and CPU registers ($p_{RAM} = 5e - 4$, $p_{CPU} = 1e - 4$).	28
3.2	Numerical results that are computed by the DEPM for the first implementation of the PID controller through the modeling of data errors in a single variable in RAM. (Expected number of failures: N_{err} , Probability of a failure: P_{err}).	30
3.3	Numerical results that are computed by the DEPM for the second implementation of the PID controller through the modeling of data errors in a single variable in RAM.	34
3.4	Numerical results that are computed by the DEPM for the third implementation of the PID controller through the modeling of data errors in a single variable in RAM.	35
3.5	Numerical results that are computed by the DEPM for the fourth implementation of the PID controller through the modeling of data errors in a single variable in RAM.	36
3.6	Comparison of the assembly code properties of the four functionally equivalent Simulink implementations of the PID controller.	37
3.7	Comparison of the evaluated reliability properties of the four functionally equivalent Simulink implementations of the PID controller for fixed data error probabilities in RAM and in CPU ($p_{RAM} = 5e - 4$, $p_{CPU} = 1e - 4$).	37
6.1	The benchmark set.	87

List of Listings

A.1 PRISM DTMC model (part 1) for the DEPM shown in Figure 2.3.	99
A.2 PRISM DTMC model (part 2) for the DEPM shown in Figure 2.3.	100
A.3 PRISM properties for the DEPM shown in Figure 2.3. . . .	100

List of Abbreviations

ACE	Architecturally Correct Execution
AT	Acceptance Test
AVF	Architectural Vulnerability Factor
CFG	Control Flow Graph
CPU	Central Processing Unit
DEPM	Dual-graph Error Propagation Model
DFG	Data Flow Graph
DTMC	Discrete-Time Markov Chain
ECC	Error Correcting Codes
EDDI	Error Detection by Duplicating Instructions
FIT	Failures In Time
LLVM	Low Level Virtual Machine
MORE	MOdel-based REdundancy
MTBF	Mean Time Between Failures
MTTF	Mean Time To Failure
NMR	N-Modular Redundancy
PID	Proportional-Integral-Derivative
PVF	Program Vulnerability Factor
RAM	Random Access Memory
SDC	Silent Data Corruption
SER	Soft Error Rate
SEU	Single Event Upset
SSE	State Space Explosion
SWIFT	Software Implemented Fault Tolerance
TMR	Triple Modular Redundancy
TPN	Timed Petri Net

Chapter 1

Introduction

Model-based design is a mathematical and visual method that can be applied in designing embedded software, such as complex control systems. Control system engineers typically prefer to implement control algorithms with MATLAB[®] Simulink[®] [46] block diagrams. For instance, in the automotive industry, most control system software is developed through the model-based design approach. Code is automatically generated from a Simulink model, and then deployed on a target embedded hardware platform.

Embedded systems are subject to hardware faults, such as bit-flips. The likelihood of the occurrence of hardware faults increases as the size of the integrated circuits decreases. Bit-flips may affect RAM cells and CPU registers in safety-critical systems and, in turn, cause data errors. These errors may propagate to critical system outputs and cause system failures. Thus, electronic devices may exhibit abnormal behavior due to bit-flips. The reliability properties of the developed control algorithm implementations (i.e. the vulnerability to hardware faults) must be numerically evaluated to assess the consequences of data errors for systems.

It is crucial to provide means by which engineers can consider the reliability properties of control systems that were developed with Simulink block diagrams at the model level. On this basis, engineers can design and evaluate reliable control systems at the Simulink model level. The design space from the dependability perspective includes i) the choice of a reliable implementation of a certain control algorithm, and ii) the choice of a suitable fault-tolerant design (mechanism) that can be applied at the model level.

First, Simulink provides a variety of design choices for implementing the same functionality, such as a certain control algorithm. However, these functionally equivalent implementations may exhibit completely different reliability properties (i.e. vulnerability to hardware faults). Given the random hardware faults that occur in a micro-controller, an assembly-level reliability assessment of model-based software, such as control system, is required. This thesis introduces an analytical method by which control system designers can evaluate the reliability properties of Simulink models at the assembly level

on a basis of a probabilistic modeling of data errors that occurred in RAM and CPU. The application of this method to the functional equivalents can identify the most reliable option (i.e. the least vulnerable to data errors).

Second, fault tolerance is significant for a dependable system design. Redundancy is the key underlying concept for achieving fault tolerance that is usually implemented at the hardware or software level. In the case of model-based development, redundancy mechanisms are preferable for direct application at the model level, such as in Simulink. In this study, a systematic classification of fault-tolerant design patterns is proposed, taken into account during the design, and applied directly at the Simulink model/control system level to tolerate random hardware faults. As one advantage, the proposed model-based redundancy mechanisms (i.e. fault-tolerant design patterns) can allow design engineers to build fault-tolerant control systems in the early model-based development phase.

Third, engineers can locate and correct systematic errors early in the system design by performing verification and validation or fault injection in the early development phase. Concerning an analytical reliability evaluation, it is more convenient for design engineers to estimate the reliability properties of Simulink models under random hardware faults and even to design reliable control systems at the model level, where the model-based design is actually applied. Thus, this thesis addresses the additional challenge of evaluating reliability properties of control systems at the model level. To this end, it introduces an analytical method for the reliability evaluation of Simulink models under data errors caused by bit-flips at the model level in an early development phase. Furthermore, it verifies the efficiency of the proposed model-level reliability evaluation of Simulink models to demonstrate that the evaluated reliability properties of control systems at the model level can represent the reliability properties that are assessed at the assembly level.

1.1 Research objectives and contributions

Control system engineers usually focus on system function and performance aspects during the design phase. In various modern applications, dependability aspects, including high system reliability and resilience, have also become key design requirements. In this context, the main aim of this thesis is to provide an easy and transparent means for a model-based dependability-oriented design at the block diagram level, such as MATLAB Simulink automated frameworks, with which control system engineers are most familiar. This

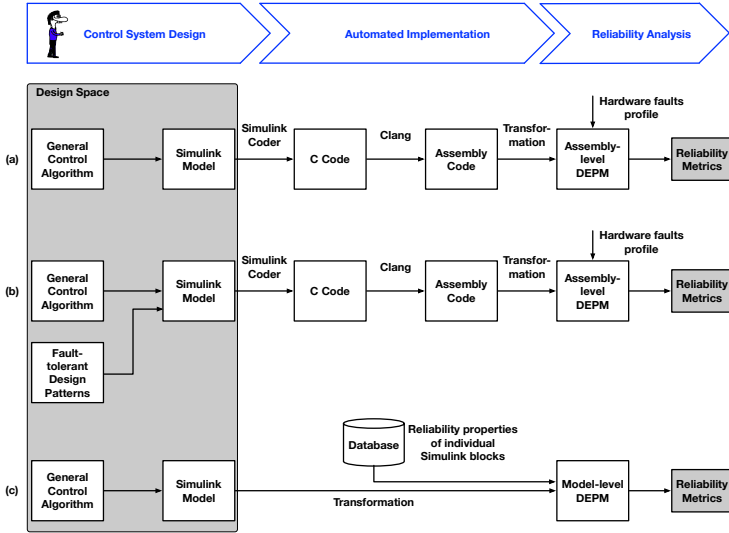


Figure 1.1: Three key contributions of the thesis.

allows control system engineers to account for dependability-related design options at the control systems model level under the support of a rigorous tool-based dependability analysis framework (see Figure 1.1).

The first objective *O.1* is to analytically evaluate reliability properties of control algorithm implementations that were developed with Simulink models to select the most reliable one. The second objective *O.2* is to provide a systematic classification of fault-tolerant design patterns that can be applied to control systems at the Simulink model level to tolerate hardware faults. The third objective *O.3* is to evaluate the reliability properties of control systems at the model level.

The main contributions of this thesis (see Figure 1.1) are listed as follows:

1. Reliability evaluation method for control algorithm implementations (Figure 1.1a): an analytical method is introduced for the overall system reliability evaluation of control algorithm implementations under data errors that occurred in RAM and CPU at the assembly level. As

a representative example, the method was applied to four common Simulink implementations of a proportional-integral-derivative (PID) controller and yielded completely different reliability metrics. The analytical method enables an early system reliability evaluation. Moreover, its application to possible implementations of a particular control algorithm helps to select the most reliable one.

2. Fault-tolerant design patterns that can be applied at the control algorithm level/model level (Figure 1.1b): a classification of fundamental, implementation-independent design patterns is proposed after the comprehensive analysis of fault-tolerant concepts. Three basic and four combined design patterns are introduced and covers most of well-proven fault-tolerant implementations. It also demonstrates how the introduced fault-tolerant design patterns can be applied directly at the Simulink level to tolerate hardware faults. A new, MOdel-based REDundancy (MORE) technique is also defined.
3. Reliability evaluation of control systems at the model level (Figure 1.1c): a method is proposed for the reliability evaluation of Simulink models under data errors at the model level, extended with the assembly-level evaluation. The efficiency of the proposed model-level evaluation method is verified by a comparison of the reliability properties that were evaluated at the assembly and model levels.

1.2 Thesis structure

The thesis is organized as follows. Following this introduction, Chapter 2 describes the preliminaries of this research, including the terminology of dependability, hardware faults (e.g. bit-flips), model-based design of control systems, and the underlying dual-graph error propagation model. Chapter 3 introduces a new analytical method for the overall system reliability evaluation of embedded control systems, which were designed with Simulink models, under data errors that occurred in RAM and CPU. Then, it details application of the method to four functionally equivalent Simulink implementations of a PID controller. The results reflect which implementation is the most reliable. Subsequently, Chapter 4 presents a classification of implementation-independent fault-tolerant design patterns. Chapter 5 describes a new model-based redundancy technique to tolerate hardware faults and demonstrates that the fault-tolerant design patterns that were intro-

duced in Chapter 4 can be applied directly at the model level, for example, a Simulink model. Chapter 6 then explains a method for the reliability evaluation of Simulink models at the model level. The related state of the art, as well as more detailed contributions of the proposed methods, are discussed individually in the respective chapters. Finally, Chapter 7 summarizes the thesis and suggests research directions for future work.

1.3 Bibliographic notes

Some parts of this thesis are based on the work that has been presented in earlier publications. An analytical method for the overall system reliability evaluation of control algorithm implementations under data errors occurring in RAM and CPU at the assembly level was published in [16], and a classification of fundamental and implementation-independent design patterns was published in [13]. Furthermore, a new, MOdel-based REDundancy (MORE) technique for Simulink models to tolerate hardware faults was published in [15]. Finally, a method for the reliability evaluation of Simulink models under data errors at the model level was published in [14].

