# Hierarchical Dependability Models Based on Markov Chains

Martin Kohlík and Hana Kubátová
Department of Digital Design
Faculty of Information Technology
Czech Technical University in Prague
Thákurova 9, 160 00 Prague 6, Czech Republic
{martin.kohlik, hana.kubatova}@fit.cvut.cz

***Abstract*—A dependability model allows calculating the rate of an event leading to a hazard state – a situation, where safety of the modeled system is violated, thus the system may cause material loss, serious injuries or casualties. A hierarchical dependability model allows expressing multiple redundancies made at multiple levels of a system decomposed to multiple cooperating blocks. A hierarchical dependability model based on Markov chains allows each block and its relation to the other blocks to be expressed independently by a Markov chain. This allows a decomposition of a complex dependability model into multiple small models to be made. The decomposed model is easier to read, understand and modify. A hazard rate is calculated significantly faster using hierarchical model, because the decomposition allows exponential calculation-time explosion to be avoided. The hazard rate of the system is the key value to specify the Safety Integrity Level (SIL).**

## I. Introduction

FPGA-based systems are sensitive to many effects that can change their programmed function [1]. These changes are most unwelcome in systems, where financial losses, serious injuries or casualties can be caused because of a failure. The improvement of dependability parameters of the final design is required to minimize the impact of such effects.

Dependability is an integrating concept including Availability, Safety, Reliability, Integrity and Maintainability [2]. We focus on the Safety parameter defined as absence of catastrophic consequences on the user(s) and the environment [2].

One of the most important design techniques improving dependability is redundancy. This means that if one part of the system fails, there is an alternate functional part. However, redundancy can have a negative impact on a system performance, size, weight, power consumption, and others [3].

There are many redundancy techniques including hardware, information, time, software redundancy, etc. [3]. We focus on hardware redundancy made by replication in this paper.

An event causing violation of safety of a system will be called *hazard event*. The rate of a hazard event is called *hazard rate*.

Dependability models – models designed to calculate a hazard rate of a system – may be created as exact or inexact models. Models of complex systems composed of cooperating modules may be created as coarse-grained – small and simple models allowing exact calculations of hazard rate in a short time. On the other hand, coarse-grained models are inaccurate and do not reflect the internal structure of the system. Fine-grained models are accurate, but they may be too large, and thus the hazard rate calculation is time-consuming. They reflect the internal structure, but they grow rapidly in size when a new module is added to a system (Cartesian product must be used).

Inexact models may be used to speed up the calculations, because accuracy is not crucial, if we prove that the inexact result is pessimistic. In other words, we must prove that the real system will be safer than the system modeled by the inexact model(s).

This paper presents a method of reducing dependability models based on Markov chains that allows us to reduce low-level models, so they contain one transition with one hazard rate only. The transition corresponds to hazard event of modeled part of the system.

The reduction allows inexact hierarchical models to be built. They use multiple linked models to reflect the structure of a system. Multi-level hierarchy may be used to describe each level of redundancy independently. The hazard rates calculated from low-level models are used in higher-level models. Higher-level models are also reduced and their hazard rates are used in top-level models.

The proposed hierarchical models allow us to

1) calculate Safety Integrity Level (SIL) [4] (Top-level model reduction),
2) determine, whether the hazard event can be tolerated/omitted safely (the hazard rate is lower than a limit value specified by SIL),
3) calculate hazard rates of systems containing multiple levels of redundancy.

Hierarchical models are composed of multiple small models, so they

1) are easier to read/understand,
2) are easier to modify/manipulate,
3) allow exponential calculation-time explosion to be avoided (the dependability parameters are calculated significantly faster).

The proposed reduction method is demonstrated on a case study system containing multiple (up to 25) identical redundant blocks configured as an N-modular redundant system (NMR) in this paper. A hierarchical model is used to illustrate the reduction method. The hierarchical model uses 2 linked models (a top NMR model and a model of the block) containing up to 19 states in total, instead of up to tens of thousands states of the exact model that would result from the Cartesian product of all models.

The results indicate that the hazard rate calculated using the hierarchical model is higher than the hazard rate calculated without hierarchy (up to 1.33 times in the case study systems presented in this paper), but the CPU-time spent on the reduction of the hierarchical model is greatly reduced (up to 40 times compared to the same system modeled by a standard non-hierarchical model).

The paper is organized as follows: Section II introduces basic reliability definitions. The reduction procedure is described in Section III and applied on the case study system in Section IV. Section V concludes the paper.

## II. Theoretical Background

The presented reduction method is intended for non-renewable Markov chains [5]. A non-renewable Markov chain contains hazard and non-hazard states. Hazard states represent situations where safety of the modeled system is violated. There are paths from each non-hazard state leading to a hazard state and there are no paths leading from a hazard state to a non-hazard state.

The failure distribution function $F(t)$ is a complementary function to reliability function. Reliability function $R(t)$ is a probability that the system will perform its intended function under specified design limits from time 0 until time $t$ at least [2], [6].

Hazard rate ($\lambda$) is defined as a constant failure rate $f(t)$ that is a conditional probability of failure density function if the failure has not occurred until time $t$ [6].

## III. Dependability Models Reduction

The reduction of a dependability model is made by joining all non-hazard states into a single state as shown in Fig. 1. The hazard state remains intact, the index ”E” (Exact) is used to mark the hazard state of the exact model. Any dependability model can be reduced using this procedure to the same reduced model shown in the lower part of Fig. 1. The reduced model contains a new hazard rate $\lambda_{Hazard}$ – the hazard rate substituting all hazard rates in the exact model. $\lambda_{Hazard}$ is the inexact hazard rate of the modeled system that has to be calculated.

The drawback of the reduction is the loss of accuracy. Accuracy is not crucial in our case, but we must prove that the inaccurate hazard rate calculated from the reduced model is pessimistic. In other words, we must prove that the real system will be at least as safe as the system modeled by the reduced model(s). The proof is made by comparison of the reduced failure distribution function $F_R(t) = 1 - e^{(-\lambda_{Hazard} * t)}$ with the failure distribution function $F_E(t)$ of the exact model. The reduced failure distribution function must be greater than the failure distribution function of the exact model all the time. This condition is called the **main requirement**.

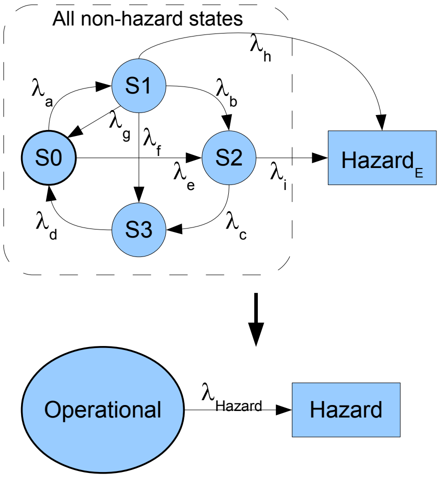

Fig. 1. Illustrative example of dependability model reduction.

The reduction is made as follows:

1) Calculate the exact failure distribution function $F_E(t)$[1].
2) Find the estimated value $\lambda_{Hazard\_Est}$.
   The main goal of this step is to make a fast estimation of the hazard rate that will be used as the starting point of the next step. This value may not meet the main requirement, but the better is the estimation, the faster will be the next step.
   $\lambda_{Hazard\_Est}$ is taken from the estimated failure distribution function $F_R(t)$ intersecting the exact failure distribution function $F_E(t)$ from the previous step at the predefined probability *level*.

$$F_E(t_{level}) = level = F_R(t_{level}) = 1-e^{(-\lambda_{Hazard\_Est} * t_{level})}$$

3) Make correction of $\lambda_{Hazard\_Est}$ to satisfy the main requirement.
   The goal of the correction is to find the lowest value of $\lambda_{Hazard\_Est}$ whose $F_R(t)$ meets the main requirement with the accuracy given by *minStep* parameter. The search is based on the bisection method – a method for iteratively converging to a solution which is known to lie inside some interval – searching for a point, where $F_E(t) \leq F_R(t)$ all the time and $\lambda_{Hazard\_Est}$ has the lowest possible value. There are three main loops (see the flowchart shown in Fig. 2):
   a) Find and verify the endpoints of the interval that will be bisected (*start* and *end*).
      - *start* is a hazard rate whose $F_R(t)$ does not meet the main requirement.
      - *end* is a hazard rate whose $F_R(t)$ meets the main requirement.

[1] Numeric method performed by NDSolve command of Mathematica 8.01 [7] software is used in this paper.

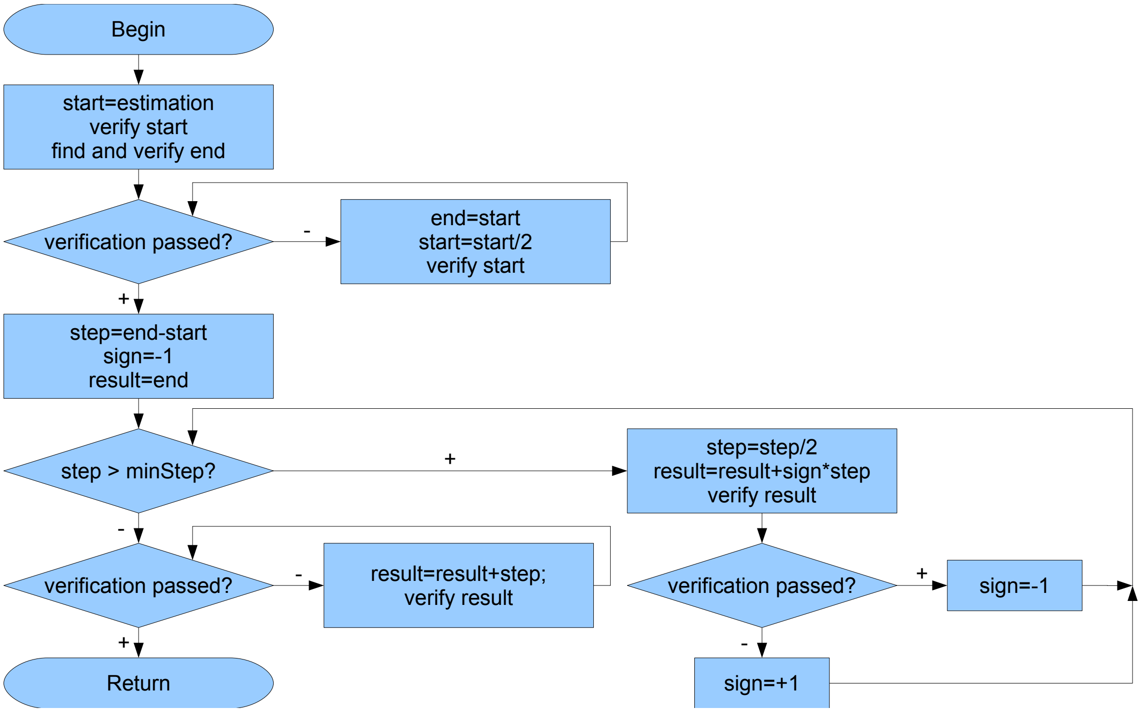

Fig. 2. Correction algorithm flowchart.

b) Perform the bisection until the required accuracy given by minStep is met.

c) Verify the result.

The flowchart shown in Fig. 2 contains subroutines *verify value* and *find end*. *Verify value* checks whether $F_R(t)$ using selected *value* as hazard rate is greater than $F_E(t)$ all the time or not. In other words, it checks whether the *value* meets the main requirement.

The *find end* method is used to find the end of the interval that will be bisected. A new function $M(\lambda)$ is created as the best-fitting function using three points – three maximal differences between $F_E(t)$ and $F_R(t)$ using three consecutive hazard rates with minStep difference. The *end* value is taken as the hazard rate, where the extrapolation of function $M(\lambda)$ reaches 0.

## IV. Case Study System

### A. Two-out-of-two Block

The safety of blocks based on Two-out-of-two redundancy cannot be violated by a single fault. The Two-out-of-two (2oo2) model is currently used as a dependability model of the railway station signaling and interlocking equipment [8].

Any railway station signaling and interlocking equipment must meet dependability requirements given by the European Standard EN 50129:2003 [4]. This standard is focused on railway equipment systems classified as the safety-critical systems. These standards define that all railway equipment systems must meet Safety Integrity Level (SIL) 4. SIL 4 means that any event whose rate is higher than $10^{-8}$ per hour must be taken into account during the dependability calculations. Any event whose rate is lower than $10^{-8}$ per hour may be neglected safely.

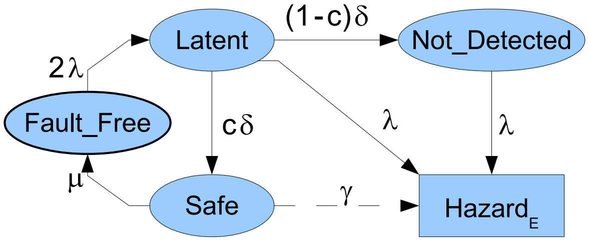

Fig. 3. Dependability model of Two-out-of-two block used to calculate the exact failure distribution function.

*Block Description*

The dependability model of 2oo2 used in this paper is constructed using the following assumptions:

- Two faults will never occur in the block at the same time.
- Assuming a fault occurs in one module, the redundant module is able to lock the block into a safe state, so that a possible future fault will not cause a hazard state. A safe state is considered as the situation where the block is not operational, but the safety is not violated (e.g. all lights are red and the traffic is operated by a human operator).
- If another fault occurs before the redundant module locks the block, the safety of the block may be violated. This double-fault situation is considered as a hazard state.

*Exact Dependability Model*

The model shown in Fig. 3 is used to calculate the exact failure distribution function $F_E(t)$ of the 2oo2 block.

$Fault\_Free$ is the functional/fault-free state of the block. The fault rate of the first fault is set to $2\,\lambda$, because the first fault can affect two functional parts of the block. $Latent$ state is active when the block contains a fault that has not been detected yet.

The self-test rate is labeled as $\delta$. If the self-test is performed successfully (a fault is detected), the block will be locked in the $Safe$ state. The probability of successful self-test is labeled as $c$.

If the self-test fails (the fault is not detected), the block will be in $Not\_Detected$ state. The safety of the block is not violated in this state, but another fault (with fault rate $\lambda$) affecting the unaffected functional part will lead to safety violation ($Hazard_E$ state). The second fault hit inside already affected functional part cannot cause a hazard, because the second functional part works correctly.

The arc leading from $Latent$ to $Hazard_E$ expresses the possibility that a second fault affects the unaffected functional part before the self-test is finished.

The block locked in the $Safe$ state waits until the repair is finished (repair rate $\mu$). The block is not functional in this state, but the safety is not violated.

The functionality of the block will be performed by a human operator, when the block is locked in the $Safe$ state. The rate $\gamma$ expresses the human operator's hazard behavior rate. This rate should be included into the safety analysis if a more complex analysis needs to be done.

The probability of detection of a fault, the fault rate and the self-test rate of the block form the following parameters values. The values have been taken from [9].

$\mu = 24^{-1}\,[h^{-1}]$ – the repair rate

$\lambda = 10^{-5}\,[h^{-1}]$ – the fault rate

$\delta = 10^{-1}\,[h^{-1}]$ – the self-test rate

$c = 0.6$ – the probability of detecting a fault by the self-test

$\gamma = 10^{-3}\,[h^{-1}]$ – the human operator's hazard behavior rate

### *Dependability Model Reduction*

The reduced model of the 2oo2 block is the same as shown in the right part of the illustrative example in Fig. 1.

The steps of reduction correspond to the algorithm described in Section III.

*Calculate the exact failure distribution function $F_E(t)$.*

The system of differential equations describing the dependability model of 2oo2 block shown in Fig. 3 can be found in [10].

*Find an estimated value $\lambda_{Hazard\_Est}$.*

We use 10 different probability levels covering the interval $(0, 1)$ to gain more precise estimation in this paper. The levels are given by

$$level_i = 1.1 - 10^{-(\frac{i}{10})},\text{ where } i = 0 \ldots 9$$

This non-linear level distribution is based on experimental observations of failure distribution functions of exact dependability models. The observations indicate that the most pessimistic estimated failure distribution function intersects $F_E(t)$ in the upper part of the interval $(0, 1)$ in most cases.

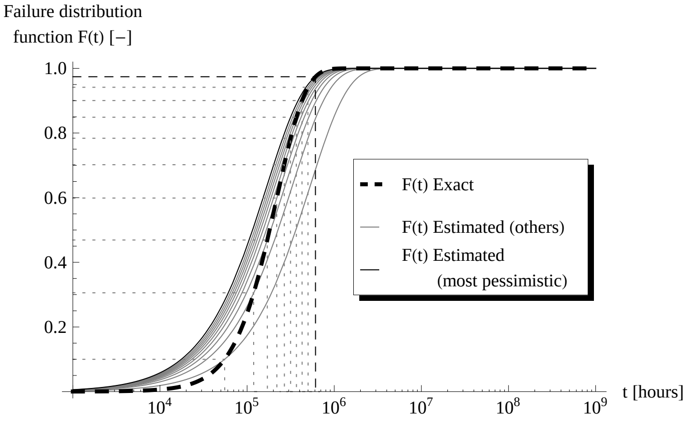

Fig. 4. Failure distribution function of the exact model ($F_E(t)$) and estimated functions intersecting $F_E(t)$ at predefined probability levels (Two-out-of-two system).

All 10 probability levels and estimations are shown in Fig. 4. The horizontal axis represents the time of operation measured in hours, the vertical axis represents the failure distribution function. The thick dashed line represents the exact failure distribution function, the black thin line represents the most pessimistic estimated failure distribution function used in the correction step, and the gray lines represent the other estimated failure distribution functions. The horizontal and vertical lines show the intersections of the exact and the estimated failure distribution functions.

The estimated value $\lambda_{Hazard\_Est}$ taken from the most pessimistic estimated failure distribution function is

$$\lambda_{Hazard\_Est} = 5.996 \times 10^{-6}\,[h^{-1}]$$

*Make correction of $\lambda_{Hazard\_Est}$ to satisfy the main requirement.*

The estimated failure distribution function from the previous step does not meet the main requirement defined in Section III, because there is an area, where the exact failure distribution function is greater than the reduced failure distribution function. The correction made according to the flowchart shown in Fig. 2 shown in Section III is necessary in such case.

The plot shown in Fig. 5 shows the exact failure distribution function, the estimated failure distribution function from the previous step and the reduced failure distribution function. The horizontal axis represents the time of operation measured in hours, the vertical axis represents the failure distribution function. The thick dashed line represents the exact failure distribution function, the gray line represents the estimated failure distribution function, and the black line represents the reduced (corrected) failure distribution function. The area, where the exact failure distribution function is greater than the reduced failure distribution function, is highlighted by a light-gray shading (see the zoom window).

The corrected value $\lambda_{Hazard\_2oo2}$ is

$$\lambda_{Hazard\_2oo2} = 7.892 \times 10^{-6}\,[h^{-1}]$$

The CPU-time[2] spent on reducing the 2oo2 dependability model is

$$t_{Reduction\_2oo2} = 11.14\,[s]$$

[2] Running on Intel Core i5 @3.3 GHz, OS: Win7 64-bit

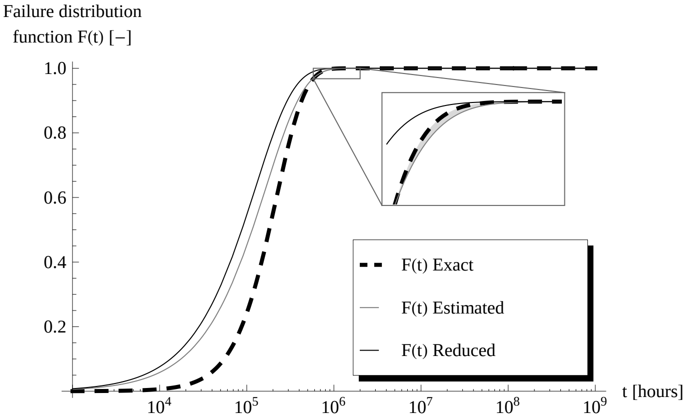

Fig. 5. Exact, estimated and reduced failure distribution functions of the Two-out-of-two system.

This CPU-time will be used in Section IV-D1 to compare runtimes of reduction of hierarchical and non-hierarchical dependability models of the system using 2oo2 as a block.

### *B. N-modular Redundancy*

The model shown in Fig. 6 is used to calculate the exact failure distribution function of a generic N-modular Redundant (NMR) system. The NMR system containing $N$ blocks will contain $\left\lfloor \frac{N}{2} \right\rfloor$ transient hazard states. These states correspond to the numbers of blocks that are in the hazard state. We reduce NMR systems consisting of 1 (a single block) to 25 blocks in this section.

### *C. NMR based on Two-out-of-two Blocks*

The hierarchical dependability model of the NMR system based on 2oo2 blocks is shown in Fig. 7. The model of a 2oo2 block is created and reduced and the result of reduction ($\lambda_{Hazard\_2oo2}$ calculated in Section IV-A) is taken as the hazard rate ($\lambda$) of the NMR model. The result of the reduction of the hierarchical dependability model ($\lambda_{Hazard}$) is calculated in the following section.

### *D. Results*

*1) Comparison of Runtimes:* Table I shows the comparison. The CPU-time spent on solving[3] the system of differential equations of the dependability model generated by the Cartesian product of the dependability models of the 2oo2 blocks configured as NMR is compared to summarized CPU-time spent on reducing the 2oo2 dependability model ($11.14\,s$ – $t_{Reduction\_2oo2}$ taken from Section IV-A) and CPU-time spent on reducing the dependability model of NMR. The reduction time includes the time required to solve the exact model, to estimate $\lambda_{Hazard\_Est}$ and to make the correction. Both models (2oo2 and NMR) are small, thus the main part of the time is spent on the corrections.

Some states of the exact Cartesian-product dependability model may be merged, because the 2oo2 blocks are identical and it is not necessary to distinguish, which blocks are in a hazard state. The model without merging grows too fast, so we generate the merged model – *Cartesian model* – directly.

TABLE I
NUMBER OF STATES AND CPU-TIMES OF SOLUTIONS OF N-MODULAR REDUNDANT SYSTEM BASED ON IDENTICAL TWO-OUT-OF-TWO BLOCKS.

| NMR blocks | No. of states before merging | No. of states after merging | NDSolve time [s] | Reduction time [s] |
|---|---|---|---|---|
| n1 (2oo2)[1)] | 5 | 5 | 0.031 | 11.14 [2)] |
| n3 | 84 | 34 | 0.078 | 22.45 |
| n5 | 1360 | 121 | 0.280 | 22.79 |
| n7 | 21824 | 315 | 0.827 | 23.15 |
| n9 | 349440 | 680 | 2.231 | 23.06 |
| n11 | $5.592 \times 10^6$ | 1295 | 5.538 | 22.34 |
| n13 | $89.48 \times 10^6$ | 2254 | 12.28 | 22.86 |
| n15 | $1.432 \times 10^9$ | 3666 | 27.69 | 22.34 |
| n17 | $22.91 \times 10^9$ | 5655 | 61.42 | 22.71 |
| n19 | $366.5 \times 10^9$ | 8360 | 119.9 | 22.86 |
| n21 | $5.864 \times 10^{12}$ | 11935 | 251.0 | 22.82 |
| n23 | $93.82 \times 10^{12}$ | 16549 | 540.5 | 23.26 |
| n25 | $1.501 \times 10^{15}$ | 22386 | 969.8 | 22.87 |

[1)] NMR containing one block is equivalent to a single 2oo2 block.
[2)] Contains time to reduce a 2oo2 block only.

The first column of Table I shows the number of the 2oo2 blocks, the second column shows the number of the states of the Cartesian-product dependability model without state merging. The dependability model without state merging would be used, if the blocks were not identical. The third column shows the number of states of the Cartesian model (after state merging), the fourth column shows the CPU-time spent on solving the Cartesian model exactly and the fifth contains the sum of the CPU-times spent on reducing 2oo2 ($t_{Reduction\_2oo2}$) and reducing NMR.

As you can see, the CPU-time spent on solving the Cartesian model grows exponentially when the 2oo2 blocks are added, but the reduction time is nearly constant, therefore the reduction will be faster when the system containing more than 15 blocks is used in this particular configuration.

*2) Accuracy:* Table II shows the difference between the hazard rates calculated using the hierarchical model and the hazard rate calculated by reducing a Cartesian model directly. The first column shows number of the 2oo2 blocks, the second column shows the hazard rate of the NMR system using reduction of the Cartesian model. The third column shows the hazard rate of the NMR system using two-level reduction of the hierarchy model. The fourth column contains the ratio of the hazard rates shown in the previous columns.

As you can see, the higher is number of the 2oo2 blocks, the higher is the error of the reduction of the hierarchical model compared to the reduction of the Cartesian model. The worst difference of the presented systems is cca. 33%, but the CPU-time spent on the reduction drops cca. 40 times.

The plot shown in Fig. 8 shows the comparison of the failure distribution functions of the N-modular redundant system based on 25 identical Two-out-of-two blocks. The reduction of this system (NMR25) is the most inaccurate of the systems calculated in this paper, because the shape of the exact failure distribution function of NMR25 system differs from the shape of the exponential function used by the reduction widely.

The horizontal axis of the plot shown in Fig. 8 represents

[3] NDSolve command of Mathematica 8.01 [7] software running on Intel Core i5 @3.3 GHz, OS: Win7 64-bit

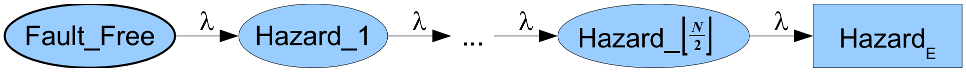

Fig. 6. Dependability model of generic N-modular redundant system used to calculate exact failure distribution function.

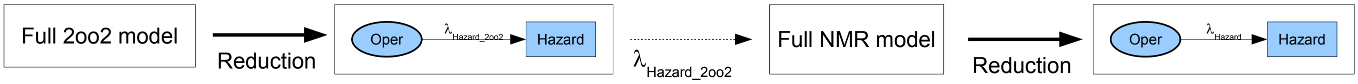

Fig. 7. Hierarchical dependability model of case study system.

TABLE II
COMPARISON OF HAZARD RATES OF N-MODULAR REDUNDANT SYSTEM CALCULATED USING HIERARCHY AND CARTESIAN-PRODUCT DEPENDABILITY MODELS.

| NMR blocks | $\lambda_{\mathbf{Hazard\_Cartesian}}$ $[\times \mathbf{10^{-6}}]$ | $\lambda_{\mathbf{Hazard\_Hierarchy}}$ $[\times \mathbf{10^{-6}}]$ | $\frac{\lambda_{\mathbf{Hazard\_Hierarchy}}}{\lambda_{\mathbf{Hazard\_Cartesian}}}$ |
|---|---|---|---|
| n3 | 14.35 | 15.20 | 1.06 |
| n5 | 19.74 | 22.43 | 1.14 |
| n7 | 24.44 | 28.04 | 1.15 |
| n9 | 28.73 | 34.12 | 1.19 |
| n11 | 32.41 | 39.22 | 1.21 |
| n13 | 35.48 | 44.00 | 1.24 |
| n15 | 38.75 | 48.53 | 1.25 |
| n17 | 41.47 | 52.30 | 1.26 |
| n19 | 43.59 | 56.45 | 1.30 |
| n21 | 46.06 | 60.50 | 1.31 |
| n23 | 48.45 | 63.75 | 1.32 |
| n25 | 50.26 | 66.87 | 1.33 |

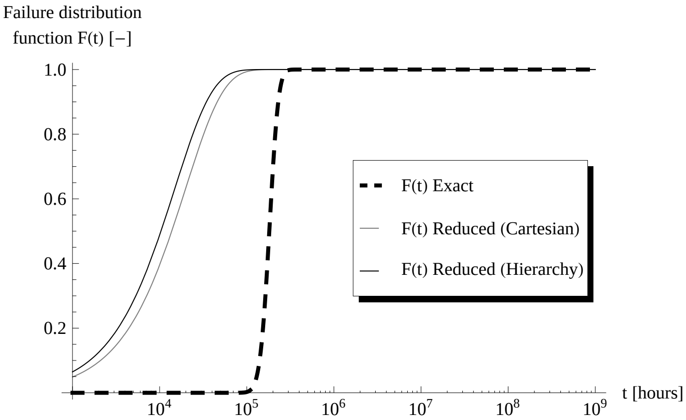

Fig. 8. Comparison of failure distribution functions of N-modular redundant system based on 25 identical Two-out-of-two blocks.

the time of operation measured in hours, the vertical axis represents the failure distribution function. The thick dashed line represents the exact failure distribution function, the gray line represents the reduced failure distribution function calculated using the Cartesian model and the black line represents the reduced failure distribution function calculated using the hierarchy model.

## V. CONCLUSIONS

A method of constructing of hierarchical dependability models based on Markov chains has been presented. The hierarchical models can be used to calculate the hazard rate – the rate of a hazard event leading to a situation where safety of a system is violated. The hazard rate is the key value specifying whether the hazard event may be tolerated or not.

The presented models are applicable to many mission-critical and safety-critical systems including railway systems, automotive systems, etc.

The proposed hierarchical model has been used to calculate the hazard rate of a complex system. The system uses two-level redundancy – Two-out-of-two method currently used in railway station signaling and interlocking equipment and N-modular redundancy. The results indicate that the hazard rate calculated using hierarchical dependability models based on Markov chains is higher than the hazard rate calculated without using hierarchy (by 33% in the worst case study system presented in this paper), but the CPU-time spent on the reduction of the hierarchical dependability models based on Markov chains is greatly reduced (up to 40 times compared to the same system modeled by a non-hierarchical model).

The results also show that the CPU-time spent on solving the system of the differential equations of the dependability model generated by the Cartesian product of the dependability models of the blocks grows exponentially with respect to the number of blocks used, but the CPU-time spent on solving the system of the differential equations of the hierarchical dependability model is nearly constant.

## ACKNOWLEDGMENT

This research has been partially supported by the project SGS12/094/OHK3/1T/18.

## REFERENCES

[1] Normand, E.: Single Event Upset at Ground Level, IEEE Transactions on Nuclear Science, vol. 43, 1996, pp. 2742–2750.

[2] Avižienis, A., Laprie, J.-C., Randell, B. and Landwehr C.: Basic Concepts and Taxonomy of Dependable and Secure Computing, IEEE Transactions on Dependable and Secure Computing, Vol. 1, No. 1, January–March 2004.

[3] Pradhan, D., K.: Fault-Tolerant Computer System Design, Prentice Hall PTR, Upper Saddle River, New Jersey 1996, ISBN 0-7923-7991-8.

[4] European Standards EN 50129:2003 – Railway applications: Communication, signalling and processing systems: Safety-related electronic systems for signalling.
http://www.cenelec.eu/dyn/www/f?p=104:110:1414646381618556::::
FSP_PROJECT,FSP_LANG_ID:13550,25 (preview only)

[5] Empirical Techniques in Finance, Chapter 2: Basic Probability Theory and Markov Chains, Springer Finance, Springer Berlin Heidelberg (2005), pp. 5–17.

[6] Hoang, P.: System Software Reliability, Chapter 2: System Reliability Concepts, Springer Series in Reliability Engineering, Springer London (2007), pp. 9–75.

[7] Wolfram Mathematica. http://www.wolfram.com/mathematica/

[8] Dobiáš, R.: Methodology of Fail-safe and Fault Tolerant System Design, Doctoral Thesis, CTU in Prague (2010).

[9] Dobiáš, R. and Kubátová, H.: FPGA Based Design of the Railway's Interlocking Equipments, In EUROMICRO Symposium on Digital System Design, Piscataway: IEEE (2004), pp. 467–473.

[10] Kohlík, M. and Kubátová, H.: Reduction of Complex Safety Models based on Markov Chains, In Proc. of the 2012 IEEE 15th International Symposium on Design and Diagnostics of Electronic Circuits and Systems (DDECS), New York: IEEE Computer Society Press (2012), pp. 183–186.