

Article

Structural Analysis of Multi-Core Processor and Reliability Evaluation Model

Sergo Tsiramua ¹, Hamlet Meladze ², Tinatin Davitashvili ³, Jose-Maria Sanchez-Sáez ⁴
and Francisco Criado-Aldeanueva ^{5,*}

- ¹ Institute of Informatics, School of Science and Technologies, University of Georgia, 0171 Tbilisi, Georgia; s.tsiramua@ug.edu.ge
² Muskhelishvili Institute of Computational Mathematics, Georgian Technical University, 0131 Tbilisi, Georgia; h.meladze@gtu.ge
³ Departments of Mathematics and Computer Science, The Faculty of Exact and Natural Sciences, Ivane Javakhishvili Tbilisi State University, 0131 Tbilisi, Georgia; tinatin.davitashvili@tsu.ge
⁴ Department of Didactics of Mathematics, Faculty of Education, University of Malaga, 29071 Malaga, Spain; jmss@uma.es
⁵ Department of Applied Physics II, Polytechnic School, University of Malaga, 29071 Malaga, Spain
* Correspondence: fcriado@uma.es

Abstract: In the present paper, the models of structural analysis and evaluation of efficiency indicators (reliability, fault tolerance, viability, and flexibility) of a multi-core processor with variable structure, equipped with multi-functional cores, are considered. Using logical-probabilistic methods, the following has been developed: models for evaluating the reliability and fault tolerance of processor cores as multi-functional elements; logical-probabilistic models of the shortest paths, flexibility, and performance conditions for successful operation of multi-core processors based on multi-functional cores; and models for estimating the reliability, fault tolerance, and lifetime of multi-core processors considering all possible states of performance. The results of the structural analysis of two-core and four-core processors and the trends of increasing the efficiency indicators of multi-core processors are presented.



Academic Editors: János Sztrik,
Yuhlong Lio, Jonathan Blackledge and
Carlos Conceicao Antonio

Received: 23 October 2024

Revised: 28 January 2025

Accepted: 31 January 2025

Published: 4 February 2025

Citation: Tsiramua, S.; Meladze, H.; Davitashvili, T.; Sanchez, J.M.; Criado-Aldeanueva, F. Structural Analysis of Multi-Core Processor and Reliability Evaluation Model. *Mathematics* **2025**, *13*, 515. <https://doi.org/10.3390/math13030515>

Copyright: © 2025 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

Keywords: multi-core processor; reliability; fault tolerance; modeling; variable structure reconfigurable system

MSC: 68M15; 93B12

1. Introduction

Currently, a fundamentally new methodology for scientific research has been formulated: mathematical modeling and computational experiment on modern computers. Using a computational experiment, large and complex problems that require huge calculations are solved. Such high performance can be achieved based on the development of parallel computational algorithms (see, e.g., [1–8], where the algorithms for parallel computing systems are developed). Along with this, there arises a problem regarding the reliability of supercomputers.

Multi-core processors are widely used in modern computers to increase performance, reliability, and energy efficiency. A multi-core processor consists of tightly connected cores integrated into a single crystal with a common cache memory. Multi-core processors are found in a variety of computing systems, including central processing units (CPUs)

of general-purpose personal computers, networking, embedded, digital signal processing (DSP), computer graphics (GPU), mobile phones, and other devices. In multi-core processors, within one cycle, each core goes through four main steps, namely, receiving instructions and data, decoding instructions (for example, where the current instruction ends and where the next one begins), execution (for example, performing arithmetic or logical operations in an arithmetic logic unit), and memorizing the received result in processor registers for quick access, which is eventually transferred to the RAM.

The number of cores in general-purpose personal computer's processors is measured in the tens of units; the number of cores for specialized chips can exceed 10,000, and the number of cores in supercomputers (i.e., clusters of chips) can exceed millions. Accordingly, the structural analysis of multi-core processor systems and the quantitative evaluation and optimization of reliability and efficiency indicators due to the substantial number of all possible states of the system belong to the class of large-scale, complex problems.

A common feature of multi-core processors is their ability to function in parallel mode, which ensures the following:

1. Performance improvement—the execution of the algorithm of a problem in parallel mode reduces the time of execution;
2. Increasing economy—in a parallel system, the operating system distributes the execution of the problem between the cores;
3. Energy efficiency—by distributing tasks between cores, it is possible to scale energy consumption;
4. Increasing reliability—when one of the cores fails, the operating system replaces this core with another core;
5. Improving fault tolerance—in case of hardware or software faults in any of the cores, the multi-core system will continue to function successfully.

In multi-core processors, core reliability can be influenced by various factors, including power consumption and temperature fluctuations at different clock frequencies, which can degrade the processor's overall reliability. To address this issue and strike a balance between energy efficiency and reliability, dynamic power management with a focus on reliability is suggested in [9]. Additionally, multi-core processors offer significant potential for optimizing energy consumption by distributing tasks across cores. As highlighted in [9], leveraging the parallelism inherent in multi-core processors can enhance not only performance but also reliability.

Ref. [10] discusses the failures of the temporary or permanent nature of individual elements in multi-core processors. A temporary failure can happen once and not be repeated again. A permanent failure is mainly due to wear of the device and has a permanent nature. The article proposes dynamic tools for checking the integrity of computing processes and a method for increasing reliability at the microarchitectural level.

In [11], in order to increase the performance and reliability of multi-core processors, the creation of a new computing environment was considered, in which, in the case of failure of a core's functional block (computing element), its function is transferred to a private or shared memory block. This process can be performed when a functional block experiences a failure (moves into an idle state), while a private or shared memory block maintains a functional state. With such an approach, it is obvious that the fault tolerance and the viability of the multi-core processor are increased. Ref. [12] discusses the reliability issues of heterogeneous multi-core processors. In the article, it was shown that even minor errors in problem processing by non-identical cores are a significant problem. To solve the problem, in the article is given the detailed investigation of each running program's reliability characteristics with further dynamical distribution of the tasks between different types of cores. Ref. [13] discusses NASA's stringent reliability requirements for cosmos

multi-core computing in terms of power, mass, and cost (expenses). If the processor core, switches, or I/O or memory ports fail,

the computing load must be transferred to other resources on the circuit. Such an approach is not perfect, since there are some failures that can bring the entire chip out of order. But in general, the proposed approach can lead to a significant increase in overall reliability, that is, the period of time during which the computer can ensure the performance of the task. The effectiveness of this approach depends on a design of multi-core chip that is optimized to minimize the probability of a single chip failure and an architecture that can effectively eliminate the failure of cores, memory, and I/O elements.

The authors of article [14] fully share the opinions and approaches of the authors of discussed articles. As in homogeneous multi-core processors, which include only identical cores, in heterogeneous multi-core processors, which consist of non-identical cores, cores with corresponding blocks are considered as multi-functional elements of a multi-core processor system, since any core can perform different types of tasks, and to increase reliability, tasks can be distributed among cores. Considering full or partial core failures, multi-core processors belong to the class of multi-state systems, whose reliability models more realistically describe the technical systems, as evidenced by the review of studies in the article [14].

The work in [15] is likely the most closely related to the approach of enhancing the reliability of multi-core processors by using their structural redundancy. According to the authors, the widespread use of multi-core processors is due to their flexibility, high performance, low power consumption, and internal redundancy. While multi-core processors offer numerous benefits, reliability concerns must be addressed to ensure their suitability for critical applications. The growing complexity and high levels of integration in these systems can compromise their reliability and safety. To mitigate these issues, the authors propose an approach that utilizes N-modular redundancy (NMR) combined with M-partitioning to improve the reliability of parallel applications on multi-core processors. A case study was conducted to validate this method, which involved two parallel benchmark applications executed with additional on-chip resources. The effectiveness of the approach was assessed using the software-implemented fault injection (SWIFI) technique.

As technology advances, the power density of multi-core chips increases, leading to the formation of hotspots that accelerate device aging and can result in chip failure. Efforts to reduce power consumption through low-power techniques often compromise the reliability of new generations of processors. To evenly distribute temperature across all cores, a proactive mechanism is needed to predict future workload patterns and their associated temperature changes, allowing preventive measures to be taken before hotspots develop. Such proactive approaches depend on precise workload prediction engines. In [16], the authors examined contemporary methods for forecasting workload dynamics and evaluated their performance. They proposed a forecasting method based on support vector regression (SVR), which effectively predicts workload behavior several steps ahead. To validate their approach, the authors used PARSEC benchmark programs on an UltraSPARC T1 processor running the Sun Solaris operating system, extracting architectural traces. These traces were then used with McPAT and hotspot simulators to generate power and temperature profiles for each core. The results demonstrated that the proposed method accurately predicts workload and power dynamics, outperforming prior forecasting techniques.

In [17], a novel method leveraging machine learning was proposed to enhance CPU scheduling in heterogeneous multi-core systems by employing a matching strategy that boosts system throughput. This scheduling approach utilizes Long Short-Term Memory (LSTM), artificial neural networks, and linear regression to predict parameters such as instructions per cycle for threads executed on different cores. The predicted values are

then applied in scheduling threads through a matching process designed to maximize instructions per cycle. The findings revealed that artificial neural networks outperformed linear regression in predicting instructions per cycle. According to the authors, the machine learning-based scheduling method enhanced the performance of heterogeneous systems by approximately 1.2 times and improves throughput by 20% compared to traditional heterogeneous scheduling techniques.

The high performance of the underlying frameworks enables the generation and collection of more data related to errors/failures, considering complex software stack configurations, in a reasonable amount of time. When dealing with large datasets related to multi-core CPU failures obtained from multiple failure campaigns, it is important to filter out parameters (i.e., functions) that have no direct correlation with the system analysis of soft errors. In [18], controlled and uncontrolled machine learning methods were proposed to eliminate irrelevant information and detect correlations between failure injection results and application and platform characteristics. This new approach provides the necessary tools to explore new and more effective failure mitigation methods.

Regarding the reliability of multi-core processors, the articles discussed here and generally published in scientific journals rarely address the analysis of the structural/functional redundancy of multi-core processors and individual core execution units, as well as the issues of quantitative reliability assessment modeling from this perspective.

In this paper, we present logical–probabilistic methods for studying the structural/functional redundancy of multi-core systems and the results of combinatorial analysis. For structural analysis, we initially construct a logical matrix and logical function of the system’s functional resources (0, 1), describing the reconfigurable multi-core system’s structure. With the help of this matrix and logical function, we investigate the system’s structure and flexibility, determine the shortest operational paths, and evaluate the conditions for its functionality. At this stage, we also determine the total number of possible system states and the proportion of those states that represent operational conditions.

By transforming the logical functions of the multi-core system’s structure using logical–probabilistic methods, we obtain a form of the logical function where logical elements can be replaced with probabilistic data, and logical operations are substituted by mathematical operations. This approach allows us to derive a probabilistic representation of estimation of the system’s reliability and perform a probabilistic assessment of reliability.

In the reliability assessment model for multi-core systems, the initial data consist not of physical elements (such as cores or blocks), but of their functional capabilities, which are represented as logical or probabilistic variables. These probabilistic data can be derived based on statistical data and using Fuzzy Logic or using the Monte Carlo method. Using these data, we construct a probabilistic matrix of the system’s functional capabilities, which describe the potential capabilities of the multi-core system.

In the reliability model of multi-core systems, we consider the following constraints: the system functions in parallel mode, with independent and non-recoverable partial failures. Additionally, in the reliability model of a multi-core system, in the case of partial failure of components, we account for an instantaneous switch to an operational mode that ensures the system’s fault tolerance and the continuation of successful functionality. While it is assumed that the reliability model includes relatively strict constraints, the combinatorial and logical–probabilistic analysis of their structure, in a static mode, enables us to determine the potential capabilities of the system and perform a comparative analysis of various models [19–21].

In future research, we will explore the dynamic processes of multi-core system operation, which can be described using a Markov model, taking time-related factors into account.

2. The Reliability Assessment Model of a Multi-Functional Element

A multi-functional element (MFE) a is an element with functional redundancy, which has the ability from the $m > 1$ number of functions assigned to the system to perform any function f at any moment t from the set of its functional resources (functional capabilities) $F_a = \{f_e | e \in [1, k]\}$, $k > 1$ and, if necessary, switch to performing another function from the set F_a [19–22].

In multi-core processors, each core at any time moment can execute a certain instruction (algorithm, assignment, task, etc.) of the program (application), which can be called a function assigned to the core at a given moment of time. Thus, in an n -core processor $A = \{a_1, a_2, \dots, a_n\}$, the task $F = \{f_1, f_2, \dots, f_n\}$ is distributed among the cores that perform these tasks in sequence, parallel, or in a mixed (sequence–parallel) mode.

The number of cores n in the processor and the number of algorithms (tasks, functions, etc.) m to be distributed between the cores and the number of functional resources k_i , $i \in [1, n]$ of each core of the processor represent the structural parameters of the multi-core processor. The value of the system efficiency indicators depends on the size of these parameters, their ratio ($n > m > k_i$), and the structure of the function distribution between the cores, which will be shown below. Depending on the number m of functions of the set F , imposed on the multi-core processor, the MFE for a given system can be functionally complete ($k_i = m$, for example, in homogeneous multi-core processors) or functionally incomplete ($k_i < m$, for example, in heterogeneous multi-core processors due to non-identical cores).

A system composed of multi-functional elements has many means of operation that can be switched to in case of failure of any element. For example, in functioning a multi-core processor, it might happen that the partial or entire failure of cores as multi-functional elements with respect to the performance of the task algorithm occurs. In the case of complete failure of the processor core, the core will be replaced with another working core, and in the case of partial failure of the core, the operating system can redistribute the task execution algorithm between the cores (i.e., implement interchangeability between the cores), which ensures the successful execution of the task by the processor. Thus, a multi-core processor is an adaptable system with a reconfigurable structure, which has the property of maneuvering (reconfiguring) to ensure successful operation.

Each core of a multi-core processor represents a multi-functional element (multi-functional element, MFE). When making quantitative assessment of its reliability, it should be considered the ability to perform any one function at any moment of time or not to perform one, two, or more functions from the set of its functional resources (functional capabilities): $F_a = \{f_e | e \in [1, k]\}$, $k > 1$.

In the article, we use $F_A = \{f_1, f_2, \dots, f_m\}$ to denote the number of tasks assigned to system A (a multi-core processor) at time moment t or the number of functions assigned to the core execution blocks. We use $F_{a_i} = \{f_1, f_2, \dots, f_{k_i}\}$, $1 < k_i \leq m$, to represent the functional capacity of each core (or block), with $F_{a_i} \subseteq F_A$. When $k_i = m$, we refer to the i -th element as a functionally full multi-functional element. When $k_i < m$, the i -th element is referred to as a functionally incomplete element.

Operability of the MFE is such a state of the element when its main parameters are within the acceptable limits, which correspond to the ability of the element to perform at least one function from the set of its functional resources.

Inoperability of the MFE is its condition when it cannot perform any of its functions from the set of functional resources.

A complete failure of the MFE is an event when an element goes from an operable state to a non-operable state.

Partial failure of the MFE is an event when the element's loss of ability to perform one or more functions does not lead transition to the state of inoperability, and the element remains in the state of operability.

The MFE's reliability assessment model should include all the operability states in which it might be appear during functioning. When partial failures occur during the operation of the MFE, the number of its functional capabilities is reduced, and when it completely exhausts the functional capability resource, a full failure state occurs (although the MFE can go from a functionally full state to a state of full failure instantaneously without going through the partial failure phase).

As an example, we consider the k -functional schematic model of reliability of the MFE, wherein the set of functional resources is $F_{a_i} = \{f_e | e \in [1, k_i]\}$ (see Figure 1).

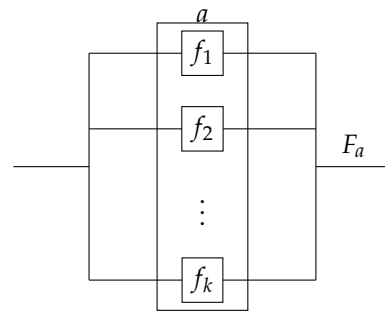


Figure 1. MFE reliability model.

The set of all possible states of the MFE (f' denotes a failure state)

$$G = \{(f_1, f_2, \dots, f_k), (f'_1, f_2, \dots, f_k), \dots, (f'_1, f'_2, \dots, f'_{k-1}, f_k), (f'_1, f'_2, \dots, f'_k)\}$$

is a set of such states wherein, at any time moment t , it is possible to have the ability to perform all functions (be in a fully functional state) or be in a state of partial failure or full failure. The number of all possible states of the k -functional MFE is $d = \text{card}\{G\} = 2^k$, from which the number of operability states will be $d^* = \text{card}\{G^*\} = 2^k - 1$.

The tree of all possible states of the MFE is shown in Figure 2, where all branches correspond to the MFE's operational state except the rightmost branch, when the MFE has lost the ability to perform all functions and is in a complete failure state.

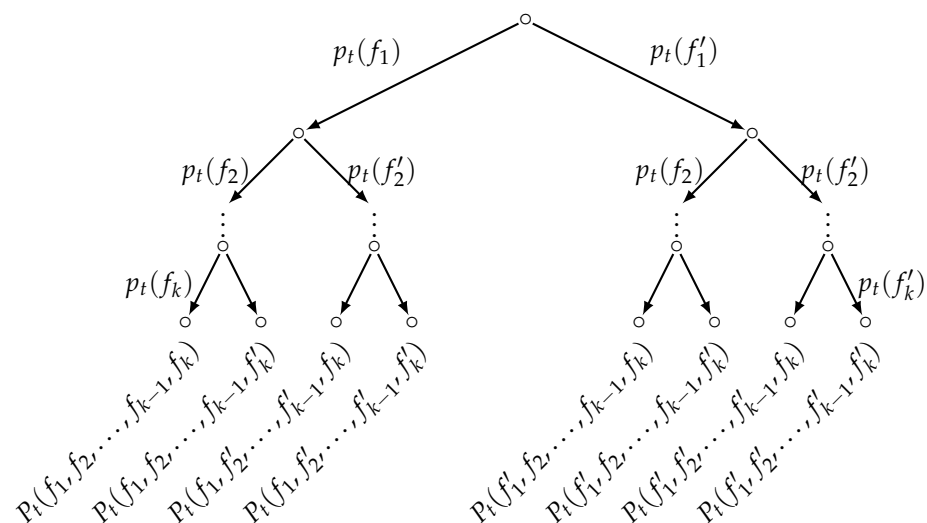


Figure 2. Tree of all possible states of MFE.

The probability that the MFE has the ability to perform all the functions assigned to the system, and that at any time t performs any one function of the set F_a , is expressed by the formula

$$P_t(f_1, f_2, \dots, f_k) = \prod_{j=1}^k p_t(f_j), \quad (1)$$

where $p_t(f_j)$ is the probability that function $f_j, j \in [1, k]$, will be executed by the element at a time moment t .

The probability that at a time moment t the MFE can perform any one function of the set F_a except the k -th function will be the following:

$$P_t(f_1, f_2, \dots, f'_k) = (1 - p_t(f_k)) \cdot \prod_{j=1}^{k-1} p_t(f_j) = q_t(f_k) \prod_{j=1}^{k-1} p_t(f_j) \quad (2)$$

The probability that the MFE cannot perform any of the functions from the set F_a at a time moment t will be the following:

$$P_t(f'_1, f'_2, \dots, f'_k) = \prod_{j=1}^k (1 - p_t(f_j)) = \prod_{j=1}^k q_t(f_j). \quad (3)$$

In the given formulas, $p_t(f_j)$ is the probability of correct (without any failure) performance of the j -th function of the F_a set at time t , and $q_t(f_j)$ is the probability of failure with respect to the function f_j .

It is clear that the sum of the probabilities of all possible states of the MFE is equal to 1. Therefore, the reliability of the MFE—the probability that at a given time t the element has an ability to perform at least one function from the F_a set, i.e., is in a functional state—will be the following:

$$P_t(F_a) = 1 - \prod_{j=1}^k (1 - p_t(f_j)), \quad (4)$$

where $P_t(F_a)$ is the probability that the MFE can operate at a time moment t , and $p_t(f_j), j \in [1, k]$, is the probability that the MFE can perform each function [22].

3. Reliability of Structurally Reconfigurable Systems

The general schematic model of the reliability of the variable structure system composed of MFEs can be presented in the following form (see Figure 3).

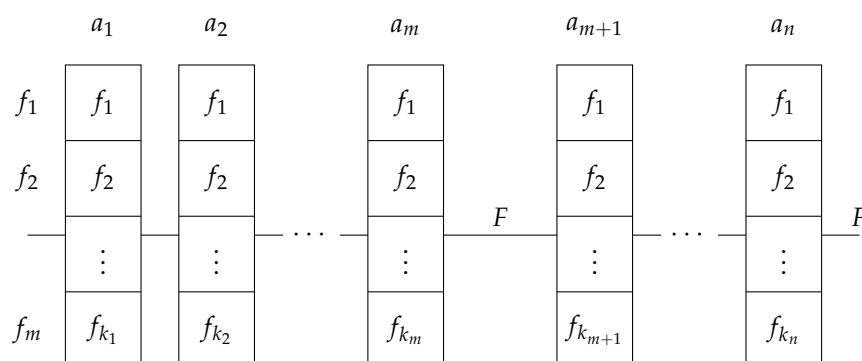


Figure 3. Reliability model of the MFE-based assembled system.

In the case of partial failure of any element in the system composed on the basis of MFEs, it is possible to redistribute the functions between the elements, which ensures the successful operation of the system. Such a possibility makes the system flexible. Ac-

cordingly, a multi-core processor based on multi-functional cores is a structurally reconfigurable system.

During system operation in a parallel mode, in case of partial failure of one of the MFEs and in order to continue successful operation, it is possible to interchange the elements, and in case of complete failure of the MFE, it is necessary to turn on the backup element. In the mode of execution of functions in order, in addition to mutual replacement, it is also possible to replace elements. That is, after performing its function, the MFE can replace another element that cannot perform the function due to it at that moment.

The structure of a system based on the MFE is defined by the composition of functional elements, their interconnections, and the distribution scheme of functions across the elements. As previously mentioned, the key parameters of the structure of reconfigurable systems are as follows:

- n : The number of cores in multi-core processor system $A = \{a_i\}, i \in [1, n]$;
- m : The number of functions (tasks, assignments, etc.) $F = (f_1, f_2, \dots, f_m)$ assigned to the system A ;
- k_i : The number of functional resources in each i -th MFE core;
- k_Σ : The total number of functional resources of all MFEs (i.e., the system's total functional resources);
- δ_S : The distribution scheme of functions between elements (structure).

Performance indicators of the system—the characteristics of reliability, flexibility, resistance to failure, and vitality—are uniquely dependent on the specified structural parameters and their interrelationship.

As mentioned above, a multi-core processor system has a flexible structure and is reconfigurable due to the functional redundancy of cores. In the event of a partial failure in any core related to its assigned function, it will be possible to replace the elements (reorganize the system structure) to restore the condition where all functions assigned to the system are performed simultaneously. It is also important to note that the efficient operation of such systems depends not only on the parameters n, m, k , and k_Σ but also on the distribution scheme δ_S of functions across the MFEs.

In order to evaluate the system's reliability (infallible operation and resistance to failure), we describe the condition of its operability.

By the system $A = \{a_i | i \in [1, n]\}$, the function $F = \{f_j | j \in [1, m]\}$ is executed successfully if at a given time moment all $n^* = m$ ($n^* \leq n$) elements complete all $f_j, j \in [1, m]$ functions from the set F , imposed on the system, so that each element at any time moment performs only one defined function from the set of its functional resources F_{a_i} .

When system A is composed of MFEs, then the function F can be performed by different distributions of functions between MFEs.

The logical function $F_A = (x_1, x_2, \dots, x_n)$, which connects the functional states of elements with the states of system operability, is called the system operability function (SMF) [23].

The matrix of functional resources of the considered system is a $(0,1)$ matrix $B(m \times n) = [a_i(f_j)]$, where

$$a_i(f_j) = \begin{cases} 1, & \text{if the employee } a_i \text{ has the ability to perform function } f_j, \\ 0, & \text{if MFE } a_i \text{ does not possess the ability to perform function } f_j. \end{cases}$$

Then, the SMF will be written using the shortest possible approach of successful operation:

$$S_q = a_{i_1}(f_{j_1}) \& a_{i_2}(f_{j_2}) \& \dots \& a_{i_m}(f_{j_m}) = 1 \quad (5)$$

where

$$\begin{aligned} i_1 &= 1, 2, \dots, n - m + 1 \\ i_2 &= i_1 + 1, i_1 + 2, \dots, n - m + 2 \\ &\vdots \\ i_m &= i_{m-1} + 1, i_{m-1} + 2, \dots, n \\ j_1, j_2, \dots, j_m &\in [1, m], \quad j_1 \neq j_2 \neq \dots \neq j_m, \quad q \in [1, N_s]. \end{aligned}$$

Here, N_s is an indicator of system flexibility, which represents the number of shortest ways for the distribution of functions between MFEs and their functioning.

The index of flexibility of the adjustable system N_s is calculated by the following formula:

$$N_s = \text{per } B(m \times n)$$

where $\text{per } B$ is a matrix permanent.

It follows from (5) that every shortest path includes the elements equal to 1 of the matrix $B(m \times n)$, which are located in the intersections of different rows and columns.

It is clear that for $n > m = k_i$, $N_s = \text{per } B(m \times n) = n! / (n - m)!$, and for $n = m = k_i$, $N_s = \text{per } B(m \times n) = n!$, $i \in [1, n]$.

The system operability condition is described by the disjunction of all shortest paths of functioning:

$$F_A[a_1(f_1), \dots, a_n(f_m)] = \bigcup_{q=1}^{N_s} S_q. \quad (6)$$

For example, the shortest approaches of functioning of the system composed of three three-functional MFEs, according to Formula (5), are written by the following conjunctions:

$$\begin{aligned} S_1 &= a_1(f_1) \& a_2(f_2) \& a_3(f_3), & S_2 &= a_1(f_1) \& a_2(f_3) \& a_3(f_2), \\ S_3 &= a_1(f_2) \& a_2(f_1) \& a_3(f_3), & S_4 &= a_1(f_2) \& a_2(f_3) \& a_3(f_1), \\ S_5 &= a_1(f_3) \& a_2(f_1) \& a_3(f_2), & S_6 &= a_1(f_3) \& a_2(f_2) \& a_3(f_1). \end{aligned}$$

According to the Formula (6), the system operability condition composed of three three-functional MFEs can be written in the form of the following disjunction:

$$F_A[a_1(f_i), a_2(f_j), a_3(f_k)] = S_1 \cup S_2 \cup S_3 \cup S_4 \cup S_5 \cup S_6, \quad (7)$$

where $i \neq j \neq k \in [1, 3]$.

The analytical form of describing the performance of a system with a reconfigurable structure composed on the basis of MFEs allows us to obtain formulas for estimating system reliability indicators using logical-probabilistic methods.

When the system is composed of MFEs, then as we mentioned above, in case of partial failure of any of the MFEs in the system, it is possible to carry out such an exchange of elements that ensures the continuation of the system's operation. The probability of such a system working without failure is determined by the sum of the probabilities of all operating states:

$$\begin{aligned} P_A(F) &= p_1(f_1, \dots, f_{k_1}) \times p_2(f_1, \dots, f_{k_2}) \times \dots \times p_m(f_1, \dots, f_{k_m}) \\ &+ p_1(f_1, \dots, f_{k_1}) \times p_2(f_1, \dots, f_{k_2}) \times \dots \times p_m(f_1, \dots, f'_{k_m}) + \dots \\ &+ p_1(f'_1, \dots, f_{k_1}) \times p_2(f_1, \dots, f_{k_2}) \times \dots \times p_m(f_1, \dots, f_{k_m}) + \dots \\ &+ p_1(f'_1, \dots, f_{k_1}) \times p_2(f_1, f'_2, \dots, f_{k_2}) \times \dots \times p_m(f_1, \dots, f'_{k_m}) + \dots \\ &+ p_1(f_1, f'_2, \dots, f'_{k_1}) \times p_2(f'_1, f_2, \dots, f'_{k_2}) \times \dots \times p_m(f'_1, \dots, f'_{k_{m-1}}, f_{k_m}), \quad (8) \end{aligned}$$

where $p_1, p_2, \dots, p_n$ are the probabilities of performing all or part of the set the functions $F = \{f_j | j \in [1, m]\}$, by the element $a_i, i \in [1, n]$, based on the functional capabilities of each of them: $F_{a_i} = \{f_j | j \in [1, k_i]\}$.

In the case when $p_1 = p_2 = \dots = p_n = p$, the probability of the system working without failure is determined as follows:

$$P_A(F) = \sum_{\gamma=0}^{k_\Sigma} N_L(\gamma) p^\gamma (1-p)^{k_\Sigma-\gamma}, \quad (9)$$

where $N_L(\gamma)$ is a number of working states of the system under conditions of the γ quantity of missing functions; the p values are the probabilities of the elements' ability to operate.

$$N_L(\gamma) = n! \left(C_{n^2-n}^{\gamma-n} - \sum_{i=1}^{n!} \frac{i-1}{i} N'_{\gamma_i} \right), \quad i \in [1, n!], \quad \gamma \in [n, n^2]. \quad (10)$$

where N'_{γ_i} is the number of mutually equivalent variants of the structure, in which the number of ways to implement the function is equal to i , and the number n of structural elements of failure-free operation is equal to γ [20].

4. Applied Methods

For the structural analysis of reconfigurable systems and the quantitative assessment of reliability criteria, we employ combinatorial analysis, sampling methods, and logical-probabilistic methods (the results are presented in Section 5).

Among the logical-probabilistic methods, we used the orthogonalization algorithm, which is based on the formula for the decomposition of Shannon's theorem:

$$f(x_1, \dots, x_n) = x_i f_1^{(i)}(x_1, \dots, x_n) \vee \bar{x}_i f_0^{(i)}(x_1, \dots, x_n). \quad (11)$$

A description of the orthogonalization algorithm and the proof of its two main theorems are provided in [23].

By using this algorithm, the logical function representing the system's operability condition initially described in its disjunction normal form (DNF) was transformed into its orthogonal disjunction normal form (ODNF). In a logical function expressed in its ODNF, each member is pairwise orthogonal (their logical product equals 0), which enables the replacement of the logical function with a probabilistic representation. Our objective was to apply an algorithm that would generate an ODNF with the fewest possible terms. To achieve this, we used two theorems, which we present without their proofs (the proofs of both theorems can be found in [23]).

Theorem 1. The negation K'_i of an r -rank elementary conjunction of $K_i = x_1 x_2 \dots x_r$, which does not contain the negations of logical elements, is equivalent to the following disjunction:

$$K'_i = x'_1 \vee x_1 x'_2 \vee \dots \vee x_1 x_2 \dots x_{r-1} x'_r. \quad (12)$$

Its members are pairwise orthogonal.

The proof of (12) is straightforward if we apply Shannon's decomposition theorem sequentially for the variables $x_1, x_2, \dots, x_{r-1}$ to the r -rank elementary disjunction:

$$D_i = x_1^{\alpha'_1} \vee x_2^{\alpha'_2} \vee \dots \vee x_r^{\alpha'_r}, \quad (13)$$

which is derived using De Morgan's formula from $K_i = x_1 x_2 \dots x_r$ elementary conjunction.

Indeed, by decomposing the logical function in Equation (13) using Shannon's formula with respect to x_1 , we obtain the following:

$$\begin{aligned} x_1^{\alpha'_1} \vee x_2^{\alpha'_2} \vee \dots \vee x_r^{\alpha'_r} &= x_1^{\alpha'_1} (1 \vee x_2^{\alpha'_2} \vee \dots \vee x_r^{\alpha'_r}) \vee x_1^{\alpha_1} (0 \vee x_2^{\alpha'_2} \vee \dots \vee x_r^{\alpha'_r}) \\ &= x_1^{\alpha'_1} \vee x_1^{\alpha_1} (x_2^{\alpha'_2} \vee x_3^{\alpha'_3} \vee \dots \vee x_r^{\alpha'_r}). \end{aligned} \quad (14)$$

Then, by further decomposing with respect to $x_2, \dots, x_{r-1}$, we obtain the Equation (12).

Theorem 2. The Boolean logical function $y(x_1, x_2, \dots, x_m)$ represented as its DNF

$$f(x_1, x_2, \dots, x_m) = \bigvee_{i=1}^n K_i (i \leq 2^m),$$

is equivalent to the function

$$f(x_1, x_2, \dots, x_m) = K_1 \vee K'_1 K_2 \vee K'_1 K'_2 K_3 \vee \dots \vee K'_1 K'_2 \vee \dots \vee K'_{n-1} K_n. \quad (15)$$

The correctness of the given transformations is easily proven by De Morgan's formulas and Shannon's decomposition theorem. If instead of each representation K'_i ($i < n$) we insert its value obtained in the form of (5), we obtain the ODNF of the Boolean function [23].

The following steps should be performed to convert the Boolean function $y(x_1, \dots, x_m)$ to its ODNF.

The validity of the given transformations is easily proven using Shannon's decomposition theorem. If we substitute the value of each expression K'_i ($i < n$) in Equation (15) with its value obtained in the form of Equation (12), we obtain the ODNF of the Boolean function $f(x_1, x_2, \dots, x_m)$. Let us describe the algorithm for transforming $y(x_1, x_2, \dots, x_m)$ into the ODNF of a Boolean function:

1. First, we need to convert the Boolean function $y(x_1, x_2, \dots, x_m)$ into DNF (disjunctive normal form).
2. Next, we renumber the DNF terms from 1 to n ($n < 2^m$) in such a way that lower-rank terms are assigned smaller numbers.
3. Using the above transformation, we define the ODNF of the function $y(x_1, x_2, \dots, x_m)$.

To reduce the number of operations, we perform the following simplifications in the conjunction $K'_1, K'_2, \dots, K'_{i-1}, K'_i$: we equate to zero those DNF terms K_j ($j \leq i - 1$) that are orthogonal to the term K'_i .

By transforming the logical representation of the system's operability condition into its ODNF form, logical elements can be substituted with element-free probabilities, and logical operations can be replaced with arithmetic ones. This transformation yields a formula to calculate the probability of flawless system operation, representing the system's reliability, as follows:

$$P\{y(x_1, \dots, x_n) = 1\} = \sum_{i=1}^s P_j(R_1, \dots, R_n), \quad (16)$$

where the orthogonal terms of the function are written in the ODNF, and R_i represents the probabilities of the truth values of logical variables x_i , $i \in [1, n]$.

To demonstrate the orthogonalization algorithm, let us consider a simple example. Let us consider a five-element logical function (Figure 4), where the reliability condition is described by the following logical function and diagram:

$$y(x_1, \dots, x_5) = y(x) = (x_1 x_3) \vee (x_1 x_4) \vee (x_2 x_4) \vee (x_2 x_5).$$

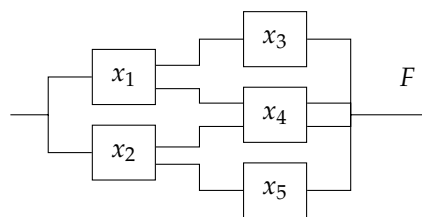


Figure 4. Scheme of a representation of a five-element logical function.

We use the following notation:

$$K_1 = x_1x_3; \quad K_2 = x_1x_4; \quad K_3 = x_2x_4; \quad K_4 = x_2x_5.$$

Based on Theorem 2, we obtain the following:

$$y(x) = K_1 \vee K'_1K_2 \vee K'_1K'_2K_3 \vee K'_1K'_2K'_3K_4, \quad (17)$$

where

$$K'_1 = x'_1 \vee x'_3 = x'_1 \vee x_1x'_3; \quad K'_2 = x'_1 \vee x'_4 = x'_1 \vee x_1x'_4; \quad K'_3 = x'_2 \vee x'_4 = x'_2 \vee x_2x'_4.$$

By multiplying the conjunctions K_i and their negations, according to Formula (17), we obtain the logical function $y(x)$ written in the ODNF (orthogonal disjunctive normal form) as follows:

$$y(x) = x_1x_3 \vee x_1x'_3x_4 \vee x'_1x_2x_4 \vee x'_1x_2x'_4x_5 \vee x_1x_2x'_3x'_4x_5.$$

In the resulting ODNF, we can replace the logical variables with probabilistic variables and logical operations with mathematical operations, which results in the following:

$$P\{y(x) = 1\} = R_1R_3 + R_1R_4(1 - R_3) + R_2R_4(1 - R_1) + R_2R_5(1 - R_1)(1 - R_4) + R_1R_2R_5(1 - R_3)(1 - R_4), \quad (18)$$

where P is the probability of the logical function $y(x)$ being true, and R_i represents the probabilities of the truth values of logical variables x_i , $i \in [1, 5]$.

When $R_1 = R_2 = R_3 = R_4 = R_5 = R$, Equation (18) takes the form of the following polynomial:

$$P\{y(x) = 1\} = 4R^2 - 3R^3 - R^4 + R^5. \quad (19)$$

By substituting $R = 0.5$ into the polynomial (19), we obtain $P\{y(x) = 1\} = 0.59375$. This exactly coincides with the reliability condition obtained using the sampling method, where the number of operability states 19 is divided by the total number of all possible states 32.

5. Discussion About Structural Analysis of Multi-Core Processors

Considering that for the ratio of structural parameters $n = m > k_i$, $i \in [1, n]$, the number of all possible states is $N = 2^{k_\Sigma}$, and for $n \geq m = k_i$, $i \in [1, n]$, the number of all possible functional states is $N = 2^{nm}$, of which the number of workable states is more than 50%, then the number of system workability states for $n > 8$ is expressed in astronomical numbers (where n , m , and k_Σ are defined in Section 3). In such a case, to obtain the system reliability estimation polynomial, it becomes necessary to use logical-probabilistic methods—orthogonalization, intersection, or recurrent algorithm-based software—and to evaluate system efficiency indicators using a computer [21].

For the structural analysis of multi-core processors, let us make a comparative structural analysis of dual-core processor A_1 with structural parameters $n = m = k_i = 2$ and

of the four-core processor A_2 with structural parameters $n = m = k_i = 4$ according to the system flexibility (N_S), reliability ($P_A(F)$), resistance to failure ($g_A(\gamma)$), and structural perfection (η_A) indicators.

The logical matrix of functional resources of a dual-core processor $n = m = k_i = 2$ has the following form:

$$B_A(2 \times 2) = \begin{vmatrix} a_1 & a_2 \\ 1 & 1 \\ 1 & 1 \end{vmatrix}$$

Based on the formula (5), the shortest ways of achieving successful functioning of the system are recorded with the following conjunctions:

$$S_1 = a_1(f_1) \& a_2(f_2),$$

$$S_2 = a_1(f_2) \& a_2(f_1).$$

The system has two ways of performing the function F ; hence, its flexibility indicator is $N_S = n! = 2$.

If we introduce the notations $a_1(f_1) = x_1$, $a_1(f_2) = x_2$, $a_2(f_1) = x_3$, and $a_2(f_2) = x_4$, the shortest ways of achieving successful operation can be written in the form of the following logical matrix (see Table 1), with the help of which it is possible to calculate the number of system performance states $N_L(\gamma)$, $\gamma \in [0, 4]$, and the total number of working states N_R .

Table 1. The logical matrix of the shortest ways of achieving successful operation in the case of $n = m = k_i = 2$.

$S_q \setminus x_j$	x_1	x_2	x_3	x_4
S_1	1	0	0	1
S_2	0	1	1	0

S_q is a way to perform the function F , where $q \in [1, n!]$, and x_j is a conditional structural element in the reliability model, $j \in [1, n^2]$.

In Table 2, for the number of failures γ , $\gamma \in [1, n^2]$, the numbers of all possible states of the system $N_M(n, \gamma)$, the number of operability states $N_L(\gamma)$, and the value of resistance to failure $g_A(\gamma)$ are given. The same table shows the value of $N_R(n) = \sum N_L(\gamma)$ and the structure perfection index η_A , which is calculated by the following formula:

$$\eta_A = \frac{N_R}{N_\Omega}, \quad (20)$$

where N_Ω is the total number of all possible states, and N_R is the total number of operability states.

Table 2. The different indicators in the case of $n = m = k_i = 2$.

γ	$N_M(n, \gamma)$	$N_L(n, \gamma)$	$g_A(\gamma)$
0	1	1	1
1	4	2	1
2	6	4	0.333
3	4	0	0
4	1	0	0
Σ	$N_\Omega = 16$	$N_R = 7$	$\eta_A = 0.375$

For a dual-core processor, where $N_R(2) = 7$, $N_\Omega(2) = 16$, the degree of perfection of the structure will be equal to the following:

$$\eta_A(2) = \frac{N_R(2)}{N_\Omega(2)} = 0.375.$$

To make the system reliability assessment model more understandable, let us consider an example of a system with structural parameters $n = m = k_i = 2$.

The schematic model of the reliability of a system composed of two dual-functional elements (a dual-core processor is operating in parallel mode), considering (5), will have the following form (see Figure 5).

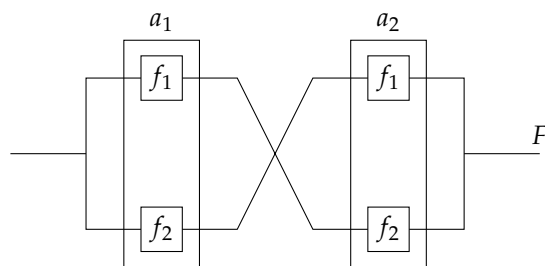


Figure 5. $n = m = k_i = 2$ system reliability model.

The general form of the polynomial to calculate the reliability of the system $n = m = k_i = 2$ will be the following:

$$\begin{aligned} P_A(F) = & p_1(f_1, f_2) \times p_2(f_1, f_2) \\ & + p_1(f_1, f_2) \times p_2(f_1, f'_2) + p_1(f_1, f_2) \times p_2(f'_1, f_2) + p_1(f_1, f'_2) \times p_2(f_1, f_2) \\ & + p_1(f'_1, f_2) \times p_2(f_1, f_2) + p_1(f_1, f'_2) \times p_2(f'_1, f_2) + p_1(f'_1, f_2) \times p_2(f_1, f'_2). \end{aligned} \quad (21)$$

As can be seen from the obtained result, in each set of the Formula (20), there are different combinations of (f_1, f_2) functions by both MFEs according to the condition of the system's operability. For the same values of probabilities of structural elements' operability, using logical-probabilistic methods, a polynomial is obtained:

$$P_A(F) = 2p^2 - p^4. \quad (22)$$

When $p = 0.99$, the estimation of structural reliability of a dual-core processor will be $P_A(F) = 2p^2 - p^4 = 0.9996$.

Now, let us consider a four-core processor with structural parameters $n = m = k_i = 4$ whose functional resource's matrix has the following form:

$$B(4 \times 4) = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \end{bmatrix}$$

The shortest ways for achieving the successful functioning of the system are given by the following conjunctions:

$$\begin{aligned}
P_F(S_1) &= a_1(f_1) \& a_2(f_2) \& a_3(f_3) \& a_4(f_4), & P_F(S_2) &= a_1(f_1) \& a_2(f_2) \& a_3(f_4) \& a_4(f_3), \\
P_F(S_3) &= a_1(f_1) \& a_2(f_3) \& a_3(f_2) \& a_4(f_4), & P_F(S_4) &= a_1(f_1) \& a_2(f_3) \& a_3(f_4) \& a_4(f_2), \\
P_F(S_5) &= a_1(f_1) \& a_2(f_4) \& a_3(f_2) \& a_4(f_3), & P_F(S_6) &= a_1(f_1) \& a_2(f_4) \& a_3(f_3) \& a_4(f_2), \\
P_F(S_7) &= a_1(f_2) \& a_2(f_1) \& a_3(f_3) \& a_4(f_4), & P_F(S_8) &= a_1(f_2) \& a_2(f_1) \& a_3(f_4) \& a_4(f_3), \\
P_F(S_9) &= a_1(f_2) \& a_2(f_3) \& a_3(f_1) \& a_4(f_4), & P_F(S_{10}) &= a_1(f_2) \& a_2(f_3) \& a_3(f_4) \& a_4(f_1), \\
P_F(S_{11}) &= a_1(f_2) \& a_2(f_4) \& a_3(f_1) \& a_4(f_3), & P_F(S_{12}) &= a_1(f_2) \& a_2(f_4) \& a_3(f_3) \& a_4(f_1), \\
P_F(S_{13}) &= a_1(f_3) \& a_2(f_1) \& a_3(f_2) \& a_4(f_4), & P_F(S_{14}) &= a_1(f_3) \& a_2(f_1) \& a_3(f_4) \& a_4(f_2), \\
P_F(S_{15}) &= a_1(f_3) \& a_2(f_2) \& a_3(f_1) \& a_4(f_4), & P_F(S_{16}) &= a_1(f_3) \& a_2(f_2) \& a_3(f_4) \& a_4(f_1), \\
P_F(S_{17}) &= a_1(f_3) \& a_2(f_4) \& a_3(f_1) \& a_4(f_2), & P_F(S_{18}) &= a_1(f_3) \& a_2(f_4) \& a_3(f_2) \& a_4(f_1), \\
P_F(S_{19}) &= a_1(f_4) \& a_2(f_1) \& a_3(f_2) \& a_4(f_3), & P_F(S_{20}) &= a_1(f_4) \& a_2(f_1) \& a_3(f_3) \& a_4(f_2), \\
P_F(S_{21}) &= a_1(f_4) \& a_2(f_2) \& a_3(f_1) \& a_4(f_3), & P_F(S_{22}) &= a_1(f_4) \& a_2(f_2) \& a_3(f_3) \& a_4(f_1), \\
P_F(S_{23}) &= a_1(f_4) \& a_2(f_3) \& a_3(f_1) \& a_4(f_2), & P_F(S_{24}) &= a_1(f_4) \& a_2(f_3) \& a_3(f_2) \& a_4(f_1).
\end{aligned}$$

The binary values x_i of shortest paths for successful system operation are given in Table 3, and the $N_L(\gamma)$ and $g_A(\gamma)$ values for different values of γ are given in Table 4.

Table 3. The logical matrix of the shortest ways of achieving successful operation in the case of $n = m = k_i = 4$.

$S_q \setminus x_i$	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	x_{10}	x_{11}	x_{12}	x_{13}	x_{14}	x_{15}	x_{16}
S_1	1	0	0	0	0	1	0	0	0	0	1	0	0	0	0	1
S_2	1	0	0	0	0	1	0	0	0	0	0	1	0	0	1	0
S_3	1	0	0	0	0	0	1	0	0	1	0	0	0	0	0	1
S_4	1	0	0	0	0	0	0	1	0	1	0	0	0	0	1	0
S_5	1	0	0	0	0	0	1	0	0	0	0	1	0	1	0	0
S_6	1	0	0	0	0	0	0	1	0	0	1	0	0	1	0	0
S_7	0	1	0	0	1	0	0	0	0	0	1	0	0	0	0	1
S_8	0	1	0	0	1	0	0	0	0	0	0	0	0	0	1	0
S_9	0	0	1	0	1	0	0	0	0	1	0	0	0	0	0	1
S_{10}	0	0	0	1	1	0	0	0	0	1	0	0	0	0	1	0
S_{11}	0	0	1	0	1	0	0	0	0	0	0	1	0	1	0	0
S_{12}	0	0	0	1	1	0	0	0	0	0	1	0	0	1	0	0
S_{13}	0	1	0	0	0	0	1	0	1	0	0	0	0	0	0	1
S_{14}	0	1	0	0	0	0	0	1	1	0	0	0	0	0	1	0
S_{15}	0	0	1	0	0	1	0	0	1	0	0	0	0	0	0	1
S_{16}	0	0	0	1	0	1	0	0	1	0	0	0	0	0	1	0
S_{17}	0	0	1	0	0	0	0	1	1	0	0	0	0	1	0	0
S_{18}	0	0	0	1	0	0	1	0	1	0	0	0	0	1	0	0
S_{19}	0	1	0	0	0	0	1	0	0	0	0	1	1	0	0	0
S_{20}	0	1	0	0	0	0	0	1	0	0	1	0	1	0	0	0
S_{21}	0	0	1	0	0	1	0	0	0	0	0	1	1	0	0	0
S_{22}	0	0	0	1	0	1	0	0	0	0	1	0	1	0	0	0
S_{23}	0	0	1	0	0	0	0	1	0	1	0	0	1	0	0	0
S_{24}	0	0	0	1	0	0	1	0	0	1	0	0	1	0	0	0

Table 4. The different indicators in the case of $n = m = k_i = 4$.

γ	$N_M(n, \gamma)$	$N_L(n, \gamma)$	$g_A(\gamma)$
0	1	1	1
1	16	16	1
2	120	120	1
3	560	560	1
4	1820	1812	0.996
5	4368	4272	0.978
6	8008	7432	0.928
7	11,440	9312	0.814
8	12,870	8010	0.622
9	11,440	4464	0.390
10	8008	1512	0.189
11	4368	288	0.0659
12	1820	24	0.0132
13	560	0	0
14	120	0	0
15	16	0	0
16	1	0	0
Σ	$N_\Omega = 65,536$	$N_R = 37,823$	$\eta_A = 0.57713$

Since $N_\Omega(4) = 2^{16} = 65536$ and $N_R(4) = 37823$, the structure perfection indicator calculated by Formula (20) will be equal to $\eta_A(4) = 0.57713$.

For a quad-core processor $n = m = k_i = 4$, for the same values of the probabilities of the structural elements' operability and using logical-probabilistic methods, the following polynomial for the estimation of system reliability is obtained:

$$P_A(F) = 24p^4 - 72p^6 - 96p^7 + 234p^8 + 528p^9 - 1808p^{10} + 2160p^{11} - 1392p^{12} + 528p^{13} - 120p^{14} + 16p^{15} - p^{16}.$$

When $p = 0.99$, $P_A(F) = 0.9999991$, which indicates the higher reliability of a four-core processor compared to a two-core processor.

If we consider the example, then a two-core processor with four operational blocks, $n = 2$ and $m = 4$, can be described by the following $(0, 1)$ matrix:

$$B_A[a_i(f_j)] = \begin{bmatrix} a_1(f_1) & a_1(f_2) & a_1(f_3) & a_1(f_4) \\ a_2(f_1) & a_2(f_2) & a_2(f_3) & a_2(f_4) \end{bmatrix},$$

where $a_i(f_j) = 1$ if the j block of the a_i core performs the f_j function, and $a_i(f_j) = 0$ in the opposite case.

If we introduce notations

$$\begin{aligned} a_1(f_1) &= x_1, & a_1(f_2) &= x_2, & a_1(f_3) &= x_3, & a_1(f_4) &= x_4, \\ a_2(f_1) &= x_5, & a_2(f_2) &= x_6, & a_2(f_3) &= x_7, & a_2(f_4) &= x_8, \end{aligned}$$

then the shortest paths for the operation of a two-core processor are written by the following conjunctions:

$$\begin{aligned}
S_1 &= x_1x_2x_3x_4, & S_2 &= x_1x_2x_3x_8, & S_3 &= x_1x_2x_4x_7, & S_4 &= x_1x_3x_4x_6, \\
S_5 &= x_2x_3x_4x_5, & S_6 &= x_1x_2x_7x_8, & S_7 &= x_1x_3x_6x_8, & S_8 &= x_2x_3x_5x_8, \\
S_9 &= x_1x_4x_6x_7, & S_{10} &= x_2x_4x_5x_7, & S_{11} &= x_3x_4x_5x_6, & S_{12} &= x_1x_6x_7x_8, \\
S_{13} &= x_2x_5x_7x_8, & S_{14} &= x_3x_5x_6x_8, & S_{15} &= x_4x_5x_6x_7, & S_{16} &= x_5x_6x_7x_8.
\end{aligned}$$

Using our application, which is based on combinatorial analysis and logical-probabilistic techniques, the following outcomes were achieved:

- For $p_i = 0.99$, where p_i represents the probability of functionality for each functional block, the processor's faultlessness probability is $P(A_F) = 0.9996$;
- The total number of all possible functional states is $N_\Omega = 2^{nm} = 2^8 = 256$;
- The number of operational states is $N_R = 81$;
- The processor's structural perfection coefficient is $\eta_A = N_R / N_\Omega = 0.31640625$.

Thus, in Section 5 of the article, we discussed two cases:

1. The combinatorial structural analysis of two-core and four-core processors, with evaluations of flexibility, reliability, and fault tolerance;
2. The combinatorial structural analysis of the functional blocks of a two-core processor, with modeling of the interchanging functions between blocks and evaluations of flexibility, reliability, and fault tolerance.

6. Conclusions

In this paper, we discuss methods for structural analysis and quantitative evaluation of reliability metrics for multi-core systems.

Models for evaluating the reliability indicators of multi-core systems have been developed, taking into account the following limitations: Only cases where cores or blocks execute tasks or functions in parallel mode were considered; the cases of independent and non-recoverable partial failures of system elements were also addressed. Additionally, in the reliability model of the multi-core system, partial failures of elements were handled by considering immediate switches to operational paths that ensure fault tolerance and the continued successful operation of the system.

The structural analysis and evaluation of efficiency indicators (reliability, fault tolerance, viability, and flexibility) of multi-core processor composed with multi-functional cores have been obtained. Using logical-probabilistic methods, the following has been developed:

- Models for evaluating the reliability and resistance to failure of processor cores as multi-functional elements;
- Logical-probabilistic models of the shortest paths, flexibility, and performance conditions for the successful operation of multi-core processors based on multi-functional cores;
- Models for estimating the reliability, resistance to failure, and viability of multi-core processors considering all possible states of performance.

Based on the comparative analysis of the structural reliability indicators of two-core and four-core processors, it was concluded that the more cores in a processor, the higher the flexibility, structural perfection, reliability, and fault tolerance indicators, and therefore, the more viable a multi-core processor is.

Author Contributions: Conceptualization, S.T.; Formal analysis, S.T., T.D. and H.M.; Methodology, validation, S.T., F.C.-A. and J.-M.S.-S.; Software S.T. and T.D.; Investigation, F.C.-A., H.M. and J.-M.S.-S.; Resources, writing—original draft preparation, S.T., T.D., F.C.-A. and J.-M.S.-S.; Data curation, visualization, S.T., H.M. and F.C.-A.; Writing-review & editing: S.T., H.M., T.D., F.C.-A. and J.-M.S.-S. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: Data are contained within the article.

Conflicts of Interest: The authors declare no conflicts of interests.

References

1. Casanova, H.; Legrand, A.; Robert, Y. *Parallel Algorithms*; CRC Press: Boca Raton, FL, USA, 2008. [\[CrossRef\]](#)
2. Bader, D.A. (Ed.) *Petascale Computing: Algorithms and Applications*; CRC Press: Boca Raton, FL, USA, 2007. [\[CrossRef\]](#)
3. Bahi, J.M.; Contassot-Vivier, S.; Couturier, R. *Parallel Iterative Algorithms: From Sequential to Grid Computing*; Chapman and Hall/CRC: Boca Raton, FL, USA, 2007. [\[CrossRef\]](#)
4. Bruaset, A.M.; Tveito, A. (Eds) *Numerical Solution of Partial Differential Equations on Parallel Computers*; Springer: Berlin/Heidelberg, Germany, 2006. [\[CrossRef\]](#)
5. Davitashvili, T.; Meladze, H. Some Algorithms of Solving the Systems of Nonlinear Algebraic Equations on Parallel Computing Systems. In *Information and Computer Technology, Modeling and Control: Proceedings of the International Scientific Conference Devoted to the 85th Anniversary of Academician I. V. Prangishvili, Tbilisi, Georgia, 3–5 November 2015*; Gorgidze, I., Lominadze, T., Khartishvili, M., Makhashvili, K., Eds.; Nova Science Publisher: Hauppauge, NY, USA, 2017.
6. Criado-Aldeanueva, F.; Davitashvili, T.D.; Meladze, H.; Tsereteli, P.; Sanchez Saez, J.M. Three-Layer Factorized Difference Schemes and Parallel Algorithms for Solving the System of Linear Parabolic Equations with Mixed Derivatives and Variable Coefficients. *Appl. Comput. Math.* **2016**, *15*, 51–66.
7. Criado, F.; Davitashvili, T.; Meladze, H.; Skhirtladze, N. On one numerical method for solving some self-similarity problems of gas-dynamics on a multiprocessor. *Int. J. Comput. Math.* **2000**, *74*, 63–85. [\[CrossRef\]](#)
8. Criado, F.; Davitashvili, T. On asynchronous iterative methods for the solution of non-linear equations and systems of equations for multiprocessors. *Parallel Algorithms Appl.* **1997**, *12*, 253–263. [\[CrossRef\]](#)
9. Waldschmidt, K.; Haase, J.; Hofmann, A.; Damm, M.; Hauser, D. Reliability-Aware Power Management Of Multi-Core Systems (MPSoCs). In *Proceedings of the Dynamically Reconfigurable Architectures*; Dagstuhl Seminar Proceedings (DagSemProc); Athanas, P.M., Becker, J., Brebner, G., Teich, J., Eds.; Dagstuhl: Wadern, Germany, 2006; Volume 6141, pp. 1–7. [\[CrossRef\]](#)
10. Kanuparthi, A.; Karri, R. Reliable integrity checking in multicore processors. *ACM Trans. Archit. Code Optim. (TACO)* **2015**, *12*, 1–23. [\[CrossRef\]](#)
11. Hajimiri, H.; Paul, S.; Ghosh, A.; Bhunia, S.; Mishra, P. Reliability improvement in multicore architectures through computing in embedded memory. In *Proceedings of the 2011 IEEE 54th International Midwest Symposium on Circuits and Systems (MWSCAS)*, Seoul, Republic of Korea, 7–10 August 2011; IEEE: Piscataway, NJ, USA, 2011; pp. 1–4. [\[CrossRef\]](#)
12. Naithani, A.; Eyerman, S.; Eeckhout, L. Reliability-aware scheduling on heterogeneous multicore processors. In *Proceedings of the 2017 IEEE International Symposium on High Performance Computer Architecture (HPCA)*, Austin, TX, USA, 4–8 February 2017; IEEE: Piscataway, NJ, USA, 2017; pp. 397–408. [\[CrossRef\]](#)
13. Villalpando, C.; Rennels, D.; Some, R.; Cabanas-Holmen, M. Reliable multicore processors for NASA space missions. In *Proceedings of the 2011 Aerospace Conference, Big Sky, MT, USA, 5–12 March 2011*; IEEE: Piscataway, NJ, USA, 2011; pp. 1–12. [\[CrossRef\]](#)
14. Yingkui, G.; Jing, L. Multi-state system reliability: A new and systematic review. *Procedia Eng.* **2012**, *29*, 531–536. [\[CrossRef\]](#)
15. Vargas, V.; Ramos, P.; Méhaut, J.F.; Velazco, R. NMR-MPar: A fault-tolerance approach for multi-core and many-core processors. *Appl. Sci.* **2018**, *8*, 465. [\[CrossRef\]](#)
16. Zaman, M.; Ahmadi, A.; Makris, Y. Workload characterization and prediction: A pathway to reliable multi-core systems. In *Proceedings of the 2015 IEEE 21st International On-Line Testing Symposium (IOLTS)*, Hersonissos, Greece, 6–8 July 2015; IEEE: Piscataway, NJ, USA, 2015; pp. 116–121. [\[CrossRef\]](#)
17. Allaqband, S.F.; Nazish, M.; Allaqband, S.F.; Bashir, J.; Banday, M.T. An efficient machine learning based CPU scheduler for heterogeneous multicore processors. *Int. J. Inf. Technol.* **2024**, 1–9. [\[CrossRef\]](#)
18. da Rosa, F.R.; Garibotti, R.; Ost, L.; Reis, R. Using machine learning techniques to evaluate multicore soft error reliability. *IEEE Trans. Circuits Syst. I Regul. Pap.* **2019**, *66*, 2151–2164. [\[CrossRef\]](#)
19. Tsiramua, S.; Meladze, H.; Davitashvili, T.; Benashvili, A. Combinatorial Reliability Analysis of Multicore Processors. *Program. Comput. Softw.* **2024**, *50*, S26–S32. [\[CrossRef\]](#)
20. Tsiramua, S.; Meladze, H.; Davitashvili, T. Logical-Probabilistic Modeling and Structural Analysis of Reconfigurable Systems Composed of Multifunctional Elements. *Pattern Recognit. Image Anal.* **2024**, *34*, 144–157. [\[CrossRef\]](#)
21. Tsiramua, S.; Sulkhanishvili, S.; Asabashvili, E.; Kvirtia, L.; Berikashvili, I. Digital Transformation of the master’s course “Structural Analysis of Systems”. In *Proceedings of the 62th International Academic Conferences*, Vienna, Austria, 20–22 May 2024; International Institute of Social and Economic Sciences: London, UK, 2024; pp. 105–112. [\[CrossRef\]](#)

22. Tsiramua, S. *System Reliability: Organizational Design of Highly Reliable Human-Machine Systems Based on Multi-Functional Operators*; Russian, Ed.; Lambert Academic Publishing: London, UK, 2017.
23. Ryabinin, I. *Reliability and Safety of Structurally Complex Systems*; Politechnika: St. Petersburg, Russia, 2000.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.