BULGARIAN ACADEMY OF SCIENCES
INSTITUTE OF INFORMATION AND COMMUNICATION TECHNOLOGIES

TASHO DIMITROV TASHEV

ALGORITHMS FOR CONFLICT-FREE SCHEDULING OF A PACKET CROSSBAR SWITCH NODE

ABSTRACT OF PhD THESIS

Supervisor:

Prof. Vladimir Monov, PhD

Sofa, 2023

The thesis contains:

- 140 pages,
- 59 figures,
- 5 tables,
- 130 bibliography sourses;

Introduction

Information technologies (IT) are used in the most diverse spheres of activity of modern society, of course - first of all - in the information sphere. They enable the optimization of various information processes and end with information modeling and forecasting of global processes in the development of nature and society

The development of IT in many aspects determines the improvement of other technologies, and allows to effectively solve the problems of informatization of the world community, taking into account the time and volume (memory) complexity of the algorithms used. Success in solving any task is mainly determined by the algorithm, the development of which to one degree or another uses a formal model with laws of composition, decomposition and properties. The main subject of the dissertation work are algorithms, which are an element of the so-called network IT - realizable in the form of local and global information systems.

Currently, telecommunication flows are digital based on packet exchange. In information exchange networks, the main nodes are switches (switch nodes), also called routers and routers. Their central unit is the switch (switching field and control circuit), which carries out the necessary "transfer" of the data packets from the input to the output communication lines by executing a calculated "schedule".The control scheme implements conflict-free transmission through the switching field following this schedule, which is calculated by a corresponding algorithm. Such conflict-free scheduling algorithms are the subject of this dissertation.

In the current dissertation, existing algorithms for conflict-free scheduling in a packet switch with a matrix switch are modeled with the means of the formal apparatus of Generalized Networks (GMNs), and an OM-model of a new algorithm called MiMa (MiMa–Minimum of Maxima) is proposed. The conducted computer simulations of the throughput of the MiMa-algorithm switch allow to determine its positive sides and also its disadvantages. This makes it possible to point out future research to improve the MiMa-algorithm.

Goals and objective of the dissertation

The purpose of this dissertation is:

To gain methodological experience in the use of the Generalized Networks apparatus in the modeling of conflict-free scheduling algorithms for a matrix switch packet switch with input buffering of the "virtual output queues" type and to propose a new algorithm and its formal OM-model for contention-free scheduling in a matrix switch packet switch. To approve a methodology for large-scale computer simulations of its throughput, providing an unambiguous comparison of different algorithms.

For this purpose, the following tasks are defined:

1. To specify models using the Generalized Nets (GNs) apparatus of classical algorithms for conflict-

free scheduling in a switch matrix packet switch, thereby gaining methodological experience from the application of the GNs apparatus to working algorithms.

2. To synthesize a new algorithm for conflict-free scheduling in a packet switch with a matrix switch and obtain its specification in the form of an GNs model.

3. To propose inbound traffic patterns realizing uniquely different loads for computer simulations of the throughput (PT) of conflict-free scheduling algorithms

4. To develop a procedure for calculating an accurate upper bound on the throughput (PS) of contention-free scheduling algorithms in a matrix switch packet switch, to unambiguously adequately compare the PS of contention-free scheduling algorithms.

Dissertation structure

The dissertation is structured in five chapters.

In the first chapter, an analytical overview of the approaches and methods for synthesis of models and information interactions and structure of complex systems is made. The necessity of using the apparatus of Generalized Networks (GNs) as a formal means of describing parallel processes is motivated. The task of calculating a conflict-free schedule in a packet switch node was chosen as a specific object for their application (GNs).

In the second chapter, the developed GNS-models for the "WaveFront" and "Observation" algorithms and their three modifications are presented. The algorithms are of sequential calculation. In this way, the methodology for selecting the minimally necessary components of the GNs for the formal specification of the algorithms for conflict-free scheduling has been worked out.

In the third chapter, a method for sequential construction of the GNs-model of the PIM-algorithm (Parallel Iterative Matching, "size" type) is described, which uses parallel computation of a conflict-free schedule. As a result, the synchronization points of the parallel processes are clearly displayed. An GNs-model of a "weighted" algorithm type - LPF - is specified. The model allows easy determination of the most "heavy" computational operation (sorting). For the purposes of computer simulation of the switch throughput, families of patterns of known types of incoming traffic (uniform, Chang, Chao, Rojas-Chessa) are defined.

In the fourth chapter, a new algorithm - MiMa - is described. It is of a "weighted" type with a sequential calculation based on a "hard" conflict criterion formulated by us. The four options for choosing weighting coefficients were investigated. The computer simulation confirms that the classical discipline "max-max" gives maximum throughput (THR), and "min-min" - minimum THR.

In the fifth chapter, a numerical procedure is specified for calculating an exact upper bound on the throughput of the switch, with an unlimited input buffer. A heuristic solution has been found for a given range of commutator field dimensions. On this basis, a comparison of THR for the PIM, MiMa and LPF algorithms was made. Proposals were made to "improve" the new MiMa-algorithm

A summary of the obtained results is presented in the **Conclusion**. Directions for future research and development are identified. A list of scientific publications on the topic and noted citations is presented.

Chapter 1 - Approaches, methods and tools for research on information interaction models

1.1 Information technologies, information processes and interactions

Information technology is the concentrated presentation of scientific knowledge and practical experience presented in a formalized form, allowing to organize in a rational way one or another frequently repeated information process [1, 2]. The most important properties of IT are described, for example, in [3]): In information systems with the application of IT, many information processes are carried out simultaneously. Those that, in order to achieve effective work, will have to interact with each other, are called interacting information processes (IIP).The general question is - which of the existing approaches to the design of complex systems would be effective for the design of RInS. And the specific - which of the existing formal methods and tools for synthesis of structures of complex objects are suitable for synthesis of VIP.

1.2 Synthesis of models of information interactions

More and more complex systems are becoming the object of modern science research. The complexity of the created systems and the traditional approaches to their design ("top-down" - top-down, composition of modules, "direct synthesis" [4]) determine one of the central problems in the theory of systems - the synthesis of their effective structures.

Characteristic properties of complex systems have been formulated [5, 6]. Distinctive features of such systems are parallelism, non-determinism, presence of interacting processes, combination of synchronous and asynchronous control, etc.

1. 2.1 Approaches and methods for synthesis of the structure of complex systems.

Currently, various approaches to the analysis and synthesis of the structures of complex systems are known. They include decomposition, coordinationand aggregation methods [7]. the methods of aggregative description of complex systems [5], the structural approach [8, 9], the approach based on the theory of complexity [10]. We are talking about a systemic approach, a structural approach Within the described design approaches, design methods are proposed, among which the following are distinguished: decomposition and aggregation, formal synthesis, synthesis based on heuristic methods, synthesis according to a generalized model [5, 11, 12 , 13].

1. 2.2 Formal means of describing parallel processes.

The purpose of the formalized description of the structure of the InS is the presentation of the available data and parallel processes in the form of special formal objects, convenient for carrying out computational and simulation experiments on a computer.

Many studies emphasize the fact that graph models are convenient and efficient means of describing and studying parallel structures and processes [8, 14, 15]. Gradually, these models were practically replaced by Petri nets (MP) [16, 17]. An interesting possibility is the application of the tensor approach of G. Kron [18, 19, 20] to the MP according to the methodology of V. Kulagin [21, 22].

However, our previous research led us to the conclusion that applying G. Kron's ideas to the apparatus of Petri nets (the methodology of V. Kulagin) does not lead to the expected results due to incompleteness in the mathematical apparatus (in the methodology of V. Kulagin). Therefore, we have chosen as the main mathematical apparatus Generalized Nets, which can be considered as a successor (upgrade and extension) of Petri nets, but have more possibilities (modeling power).

1.3 Generalized Nets (GNs).

Generalized nets are a formal apparatus designed to represent in detail the relationships between structure and temporal correspondences in parallel processes. The concept was originally presented in [23]. They are used in modeling the processes of a wide range of systems, objects and models [24], expert systems [25], machine learning [26].

The main elements of the GN are transitions - modeling actions and positions - modeling states. Positions contain kernels that go through transitions according to logical conditions. Cores have the necessary characteristics that can change when passing through transitions. The necessary theory and technique are described in [27, 28]. Of course, every Generalized Net model has a graphical representation, but it only carries part of the information about the GN model (unlike Petri nets).

After accepting the proposed formal apparatus, the practical question is: what part of all elements but GNs are needed for a specific (engineering) task. In our experience, this is a question with an informal answer. In the sense - for the specific task, such a minimal set of parameters is used, which is sufficient for the specification of the task with GN[29].

A new review of studies using GNs covering 272 titles was published in 2021 [30]. My work is also cited there. GNs are used in a variety of fields. The author of GNs K. Atanasov himself continues to actively use them - we will mention 3 publications in MATHEMATICS (MDPI, Basel, Switzerland,) [31, 32, 33].

Let us describe for what purpose and task we will use the GNs apparatus.

1.4 The conflict-free scheduling problem of a matrix switch packet switch.

Scientific and technical progress allows increasing the number of subscribers connecting to global information networks and the provision of new services. This requires new features and new gigabit/terabit routers and switches. At high speed, switching nodes have complex management for differentiated and integrated services, different interfaces, protocols, packet formats, etc. Networks must have high-speed transmission, switching and control equipment for large capacity to ensure: high reliability; great security; dynamic control [34].

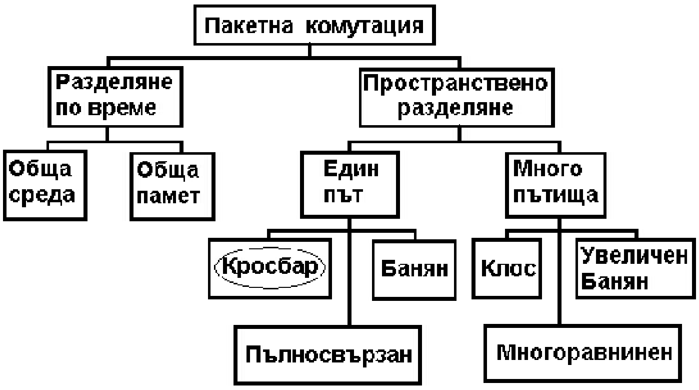
Accordingly, the packet switching architecture is moving from time-division to space-division [35], shown in Figure I.1.

In this work, we are interested in crossbar - matrix switch commutator. To transmit the maximum number of requests, parallelism is used in the switching field. This is solved by "building" the so-called conflict-free schedule (BR). Crossbar can be with input; outgoing; input-output or internal buffering [36].

A lot of attention has been constantly drawn to the possibilities of "virtual output queue" (VOQ - virtual output queue), figure I.2. At each input, buffers are mapped for each output [37]. For the management of this transmission through the switching field, the so-called conflict-free schedule (BR) is calculated. Many algorithms have been proposed for this purpose, and research is ongoing.

When creating a schedule for a switch, the goal is not only to transmit the maximum amount of packets per unit of time through it, but also to minimize the time for packets to wait, and also to minimize the probability of packet blocking.

Simultaneously achieving these three goals (mathematically, the problem is known as b-partite graph) leads to problems with non-polynomial solution complexity (NP-complete) [38] and available algorithms can partially solve this problem. That is why new proposals are appearing, also related to new technological possibilities (FPGA) - for example [39]. The last general overview of the problem can be found in [40]. As the most recent studies, we can point to [41, 42, 43, 44, 45].



Фигура I.1. Класификация на архитектурите за пакетна комутация [35]

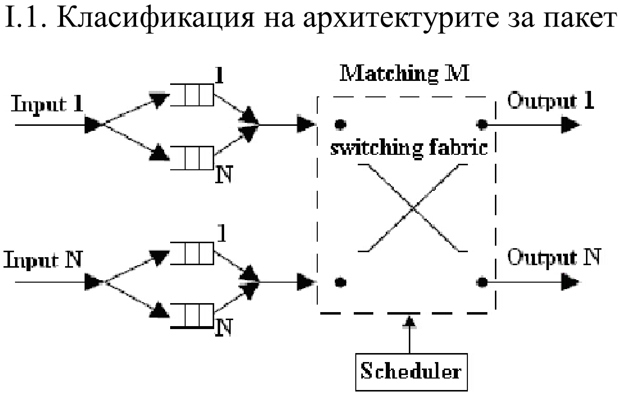


Figure I.2 Logical structure of a switch with VOQ buffers [37]

Various formalisms have been used as formal means in the description and study of the characteristics of algorithms. In our research, we use the Generalized Networks apparatus [28].

1.5 Conclusions

Information technologies for their progress need developed formal means for specification, analysis and synthesis of complex systems with parallel processes. Generalized networks are a powerful modern formal apparatus suitable for this purpose. Conflict-free scheduling algorithms in switch-matrix packet switches are an area of research that is concrete, promising, and needed.

Validation of the effectiveness of new BR algorithms begins with modeling the throughput of a switching node with a uniform traffic load. Performance check for uneven traffic follows. We need families of hardware and software independent traffic simulation models. It is necessary to synthesize models of well-known algorithms with means of OM. We want a new algorithm for BR. The proposed models for the study of traffic with different load loads should be used in computer simulations to clarify its strengths and weaknesses.

Chapter 2 - Generalized-Nets Models of Algorithms with Input Buffering and Virtual Output Queues (VOQ)

As stated, from a mathematical point of view, the bipartite graph problem is non-polynomial complexity (NP-hard) [38]. Therefore, various conflict-free scheduling algorithms have been proposed, which approach the ideal to varying degrees.

The efficiency of the operation of a switch is primarily assessed by the realized throughput (PS - throughput). Incoming packet traffic is divided into two types - uniform (uniform load traffic) and non-uniform (non-uniform) [35]. These types are further divided into balanced, asymmetric, non-balanced, bursty, and other sub-types of traffic (hot-spot, Pareto) [34, 40]. With this diversity, the question of the adequate comparison of the results arises, since the publications often do not specify the specific parameters of the traffic models used in the computer simulations.

We will start using the GNs apparatus by specifying the intuitively "simplest" algorithm "WaveFront"[34]

2.1. OM-model of a centralized (sequential) "rib" algorithm "WaveFront"

Inputs to the algorithm are packet-switched requests. Packet transmission requests (at VOQ) through an n x n switch are described by a traffic matrix T of dimensions n x n. Each of its elements tij, tij ∈ [0,1] represents a request to transmit packets from input i to output j [36]. T is formed by the row vector of the queries of all inputs.

A conflict situation is created when, on any row and/or column of the traffic matrix, the number of units is greater than one. Conflict avoidance is directly related to the efficiency of the communication node.

The presented algorithm computes a set of matrices that have only one element 1 (if any) in each row and column. This set of matrices defines a contention-free schedule for the transmission of requests in the switch.

2.1.1 GNs-model of the "Wave Front" algorithm

The task is to build a model of the algorithm (given in the full text) by means of GNS. The proposed solution in the form of GNS is shown in Fig.II.1.

The capacity of all arcs is equal to one. GNs does not have local and global time components. The analysis confirms the receipt of a conflict-free schedule. But from the chosen (deterministic) way of traversing the matrix M, priorities appear in the execution of requests.

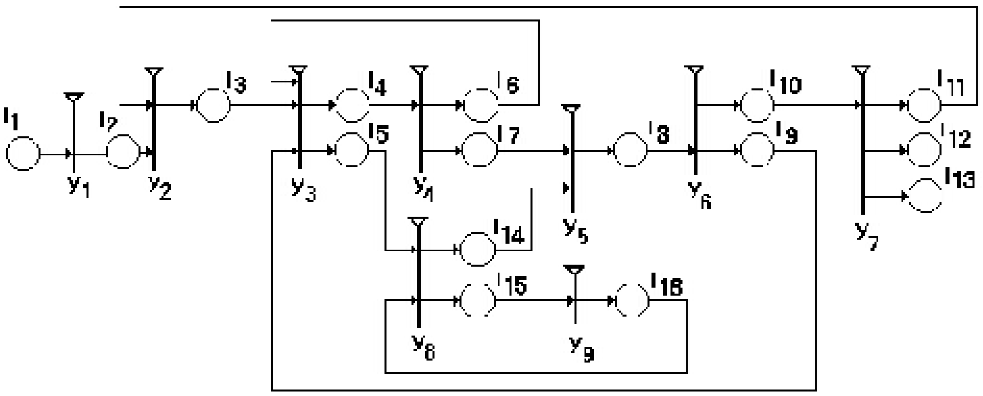


Figure II.1. GNs-model of the "Wave Front" algorithm

2.2 Two modifications ("extensions") of the GNs-model of the "WaveFront" algorithm.

We will use the GNs apparatus to specify two modifications of the algorithm by changing the discipline of traversing the columns of the traffic matrix. In the first modification ("left-right"), there is a discipline of traversing the columns of the traffic matrix "right-to-left" for even rows. In the second modification ("matrix"), when calculating the odd solutions, a discipline for the pillars is used as the main algorithm. For even solutions, we use a "right-to-left" discipline. The algorithm terminates when there are no queries left in matrix T.

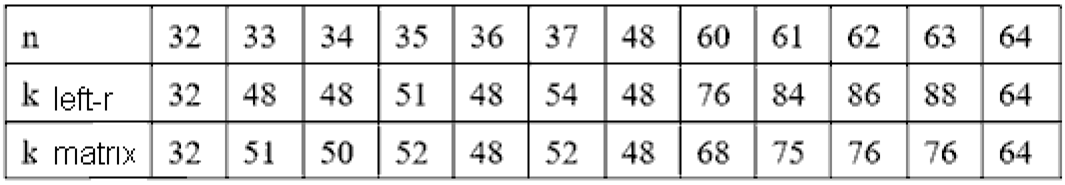
2.2.1 GN-model of the first and second modification of the "Wave Front" algorithm

In the main text, the GN-specifications of the two variants of "Wave Front" ("left-right" and "matrix") are given. A comparison of the second modification with the first, in which rows are checked sequentially in a different direction, is shown in Table 1.

When we have full traffic (T, ti,j=1), in the case when n = 2k + 2k-1, k∈N (k=1,2,3,…), all three algorithms work optimally (100% throughput ability - THR). In other cases, the PS drops sharply.

The presented first modification of the algorithm has a better PS than the second modification in the first half of the interval if n ∈ [2k,2k + 2k-1], k∈N (k=1,2,3,…), and by - bad - in the second. That is, the benefits of modifications cannot be unambiguously declared in relation to the THR..

Table 1. Comparison of traffic matrix occupancy



n	32	33	34	35	36	37	48	60	61	62	63	64
k left-r	32	48	48	51	48	54	48	76	84	86	88	64
k matrix	32	51	50	52	48	52	48	68	75	76	76	64

Table 1 shows the number k of the calculated matrices (Q) as a function of the optimal number (n) of filling the traffic matrix T (number of requests =n^2)

2.3 GNs-model of a centralized (sequential) "rib" algorithm "Observation".

This algorithm is the "simplest" possible - an analogue of Time Division Multiplexing [36]. The point is to choose for solution the elements of the main diagonal of T and its parallel diagonals.. The formal description with GNs to obtain a conflict-free schedule is given in the main text. The proposed solution in the form of GNs is shown in Fig. II.2. Main benefit - minimal amount of memory is used.

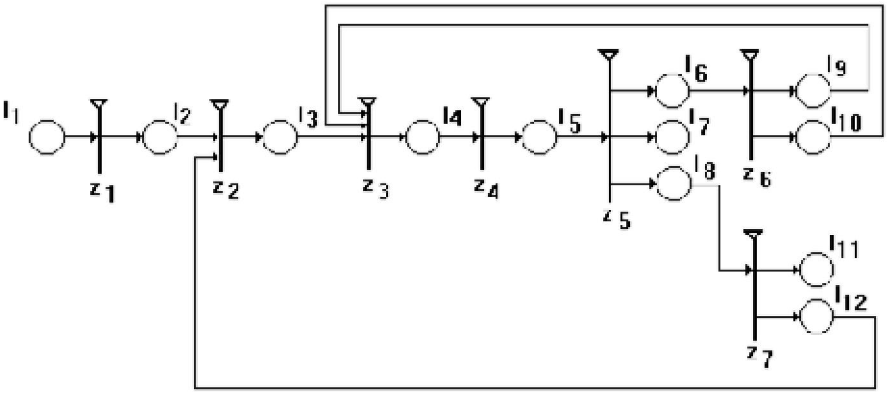


Fig. II.2. GNs-model of the "Observation" algorithm

The capacity of all arcs is equal to one. GNs has no local and global time components. The selected discipline introduces priorities in the service of requests. The algorithm is effective only when the input lines are fully and evenly loaded. Therefore, we will make a modification to the discipline of choice.

2.4 Modification of the GNs-model of the "Observation" algorithm.

The essence is instead of selecting requests from the diagonal of T ("hard" discipline), in the next row we select the element from the pillar next to the right of the already selected pillar (if there is a request). The full formal description is given in the dissertation.

The formal description with GNs is given in the main text. Computer modeling was performed using the Vfort software package of the Institute of Mathematical Modeling of the Russian Academy of Sciences [46].

The deterioration of THR-result – a larger number of matrices Q with resolved non-conflict connections is similar to the basic variant.. The two algorithms optimally solve the task only at the maximum request load. At 50 percent filling of the traffic matrix with requests, the two algorithms are approximately equal with the THR getting worse

2.6 Conclusions

A model of the "Wave Front" algorithm was built using the formal apparatus of Generalized Networks (GNs). Analysis of the pattern indicates the emergence of priority in the servicing of switching requests from different inputs. This is not a desirable property. The "Monitoring" algorithm is good (100% PS) only at full input load (which is not enough) and its modification does not help.

Using the GNs-models, a computer simulation of the throughput should be performed, in which the time complexity of the algorithm execution should be determined at the same time.

As a result, the following conclusions were drawn:

1. It is shown that Generalized Networks (GNs) can be used to model the set tasks, because they have the ability to model both the structure of the studied object and the dynamics of the processes in it.
2. For the specification of algorithms for conflict-free scheduling, only a part of the formal parameters of the GN apparatus is needed.
3. The specified algorithms, as representatives of an intuitive-obvious approach, do not have good throughput(THR).

Chapter 3 – Generalized Nets Models and the computer simulation of the efficiency of a conflict-free scheduling slgorithm

In this chapter, the GNs apparatus is used to synthesize a model of the well-known Parallel Iterative Matching (PIM) algorithm [34], in which parallel processes are explicitly specified during transmission in a packet switch. This algorithm has served as the basis for many algorithms for obtaining a switching schedule with explicit parallelism. It is also modeled with GNs and the LPF algorithm. Thus, we have representatives of the two classes of algorithms - "edge" and "weighted", and we can study their characteristics and compare them.

3.1 Types of incoming traffic. Inbound pattern families with inbound stacking.

In the THR studies of the conflict-free scheduling algorithms we model, we use a "simple" uniform input traffic template (i.i.d. Bernoulli uniform [35]). Its type is shown in Fig.III.1, we denote it as Uni1. We can perform a simulation with a "heavy" incoming traffic template (we'll call it "Pythagorean" type - the sum of each row/column is n2). Its appearance is shown in Fig. III.2., and its convergence rate proved to be very good. But the cost turns out to be an n times increase of the simulation time (where n x n is the dimension of the matrix switch).

$$\begin{matrix} 1 & 1 & \cdots & 1 \\ 1 & 1 & \cdots & 1 \\ \vdots & \vdots & \ddots & \vdots \\ 1 & 1 & \cdots & 1 \end{matrix} \qquad \begin{matrix} 1 & 2n-1 & 2n-3 & \cdots & 3 \\ 3 & 1 & 2n-1 & \cdots & 5 \\ 5 & 3 & 1 & \cdots & \vdots \\ \vdots & 5 & 3 & \cdots & 2n-3 \\ 2n-3 & \vdots & 5 & \ddots & 2n-1 \\ 2n-1 & 2n-3 & \vdots & \cdots & 1 \end{matrix}$$

Fig.III.1 "simple" pattern Uni_1 Fig.III.2 "heavy" "Pythagorean" pattern

Using the information that a sufficient condition for the "triggering" of a region of instability is that the main diagonal in the traffic matrix has zero values (for non-uniform traffic proposed by Chang [36]), we come to the conclusion that "Simple" pattern with a "zero" main diagonal is a Chang-model solution. We will refer to such a pattern as Chang_i (or “1-0”).

And in the case of non-uniform traffic, the sums of the elements of the matrix T by rows/columns must be equal [36]. In our computer simulation, we will use a necessary number of matrices T such that the total number of packets in each row and column of matrix T is equal. We will call these matrices a pattern family for the respective traffic type. They have the following properties:

- easy generation for any size of switch (*n x n*);
- generation does not depend on the type of hardware, compiler and operating system used;
- their exact, optimal, conflict-free schedule is known.

A family of patterns based on the hotspot traffic model [36], which we call Chao-model, is proposed. This model is given by: $\lambda_{ij}=0{,}5\rho$ for $i=j$, and $\lambda_{ij}=0{,}5\rho/(n-1)$ in all other cases, $i,j\in 1,\dots,n$, where ρ is input load.

The first matrix type in the pattern family is called $Chao_1$. Its optimal schedule requires $2(n-1)$ switches of the $n \times n$ switching matrix. In the general case, the i-th matrix is denoted as $Chao_i$. Its optimal schedule requires $2i(n-1)$ switches. These types of matrices are shown in Figure III.3.

$$T=\underset{2\times 2}{\begin{bmatrix}1&1\\1&1\end{bmatrix}}\ \underset{3\times 3}{\begin{bmatrix}2&1&1\\1&2&1\\1&1&2\end{bmatrix}}\cdots\underset{k\times k}{\begin{bmatrix}k-1&\dots&1\\\vdots&\ddots&\vdots\\1&\dots&k-1\end{bmatrix}}\cdots\qquad T=\underset{2\times 2}{\begin{bmatrix}i&i\\i&i\end{bmatrix}}\ \underset{3\times 3}{\begin{bmatrix}2i&i&i\\i&2i&i\\i&i&2i\end{bmatrix}}\cdots\underset{k\times k}{\begin{bmatrix}(k-1)i&\dots&i\\\vdots&\ddots&\vdots\\i&\dots&(k-1)i\end{bmatrix}}\cdots$$

Fig.III.3 Matrices of types $Chao_1$ and $Chao_i$

3.2 Generalized Nets model of PIM-algorithm

This study presents the results of a computer simulation of the generalized network model of the well-known PIM-algorithm (Parallel Iterative Matching) [34]. In it, parallel processes are explicitly specified during the operation of the packet switch.

3.2.1 Description of the PIM-algorithm

he PIM algorithm computes a series of conflict-free matrices Q_k, each of which passes through three phases.

1. Each input sends a Request to each output for which there is a packet to transmit.

2. Each output randomly selects one of the received requests and reports (Grants) it to the corresponding input.

3. Each input that received grants randomly selects only one of them. This package will be sent for transfer (Accept) [34].

In each phase there is parallelism - a direct invitation to use the GNs apparatus. This parallelism is our basis for implementing GNs.

3.2.2 Building a generalized network model of the algorithm

In the main text, the GNs model is formally described. The described three phases of the algorithm lead to no less than three transitions in the GNS-model. The graphical representation is shown in Figure III.4.

The most interesting part of the design of the GNs-model is the need to introduce the additional position p. In this way, we achieve synchronization of the three phases of the algorithm (positions p, q, out). Such a result can be seen as a consequence of the strict formal apparatus of GNs.

Each of the transitions has the same priority. Analysis of the model using shows that it computes a conflict-free schedule.

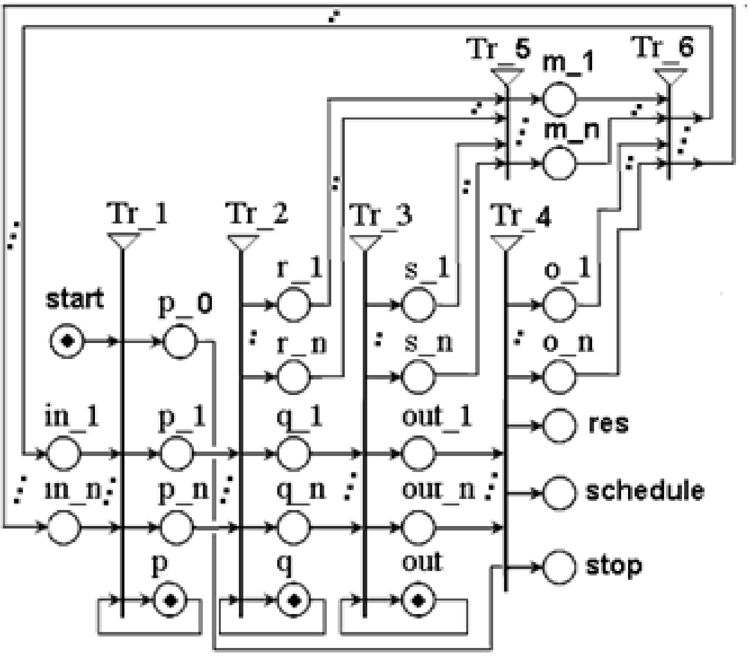


Fig.III.4 Graphic representation of the final GNs-model of the PIM algorithm

3.3 Computational experiments

The transition from GNs-model to an executable program is implemented using the VFort package [46]. An IBM compatible Pentium IV computer with a frequency rate of 3000 MHz and 2 GB RAM was used in the computational experiments.

From the performed simulations, it follows that for a "quick" evaluation of the THR of new algorithms, the "Simple" pattern Uni is satisfactory. But to assess the existence of a region of instability, $Chang_i$ should be used.

The simulation results with the starting patterns Uni_1 and $Chang_1$ are shown in the main text. Time differences are negligible. THR differences can also be neglected. PIM-algorithm has no region of instability.

In simulations with increasing size of input buffers (pattern index i), using a personal computer is a limiting factor. We sought access to greater computing power. The transition from OM-model to an executable program is implemented using the VFort package [46]. The source code is compiled using the grid cluster at CERN (http://lxplus.cern.ch), and the resulting code is executed locally on the same. The allowed execution time was 240 hours. Thus we simulated THR for PIM up to Uni_{8000} , and found it to be tending towards the known theoretical limit for THR of 63,2 %. For control, we conducted simulations on the grid structure BG01-IPP of IICT-BAS (www.grid.bas.bg). With results matching, our grid turned out to be about 10% faster.

For THR with the $Chao_i$ pattern family, we performed simulations on the grid structure BG01-IPP of IICT-BAS. In the figures below, $Chao_i$ is marked as C-i for i=1,2,… Figure III.6 shows the result of computer simulations of the PIM-algorithm with input data $Chao_1$, $Chao_2$,$Chao_5$

and *Chao*$_{10}$, and also for indices 20, 60, 100. The resulting throughput is averaged over 10,000 simulations for each size, except for *Chao*$_{10}$ and larger indices, where the average of 1000 simulations is taken.

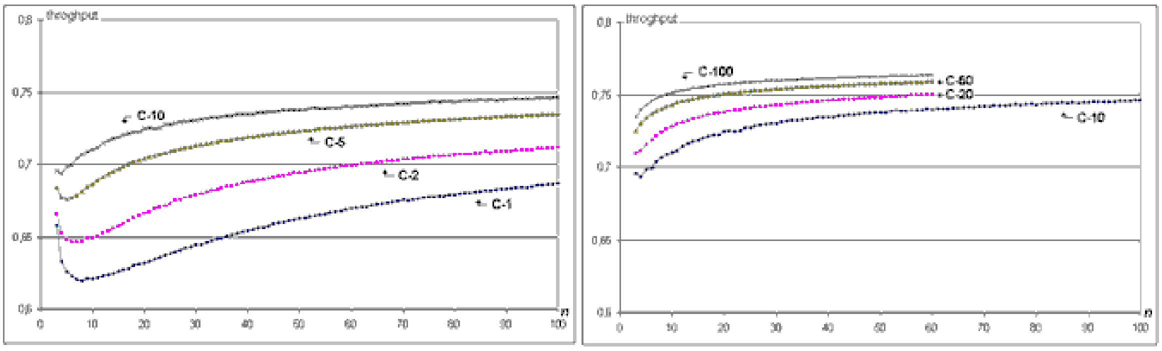


Fig.III.6 . PIM THR results for*Chao*$_{1,2,3,10\ ,20,50}$ и *Chao*$_{100}$

As shown in Fig.6, the speed of approaching the specified upper limit increases. Clearly, this limit is less than 100% and greater than 76%.

3.4 A generalized-network model of a "weight" LPF algorithm.

For the LPF-algorithm [40], the existence of an upper limit of THR is confirmed for all cases of admissible traffic (THR tends to 100%).The author of LPF presents a mathematical study, according to which LPF - is optimal in two respects [40]. Therefore, we chose the LPF algorithm for our study.

A description of the LPF-algorithm is given in the full text. The THR of the algorithm follows from the choices made in Step 5. If more than one element with the same maximum weight appears, the choice between them is made randomly [40].

3.4.1 Generalized-network model of the LPF-algorithm

The graphical representation of the GNs-model is shown in Fig.III.7. The priority of each of the transitions is the same and they have the same relation to the cores. Analysis of the model using GNs shows resolution of the conflict-free problem. The model provides additional information (k,r, etc.) that can be used to calculate the mean THR. There are options for other information as well.

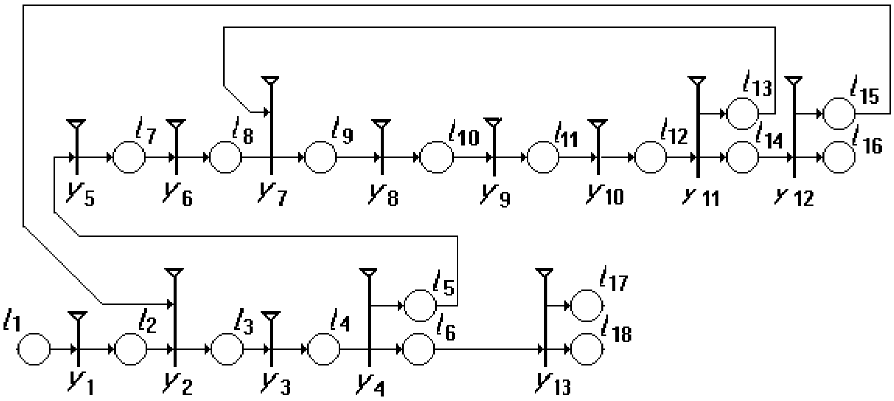


Fig.III.7. Graphic representation of the GNs-model of the LPF-algorithm.

.3.4.2 Computer simulation

We use the GNs-model to write a computer program with the Vfort package (free access [46]). Compilation is carried out using the AVITOHOL supercomputer of IICT-BAS (www.iict.bas.bg). The binary code is executed locally on AVITOHOL. The operating system is Red Hat Linux. The used resources reach 16 processors, 16 GB of RAM. The execution time does not exceed 240 hours.

Inbound traffic – a family of uniform Bernoulli distribution patterns are used to simulate uniform traffic with ρ=100%. (Uni-i). The dimension n varies from 3x3 to 44x44. 1,000,000 simulations are run for patterns Uni-1, Uni-2, Uni-3, Uni-4. Figure III.8 shows the resulting throughput (average) - on the left, and the time for one simulation on the right.

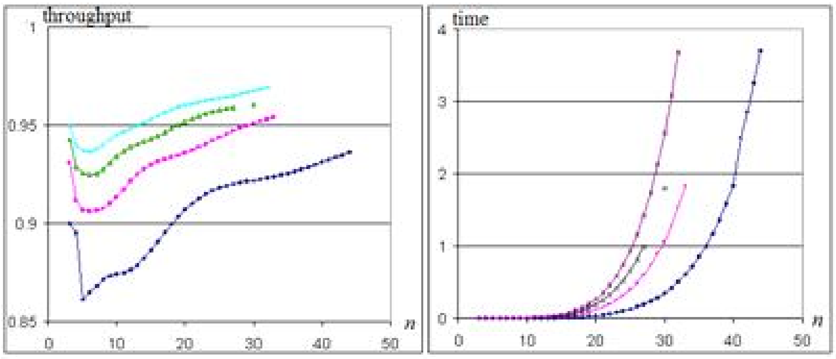


Fig.III.8 Throughput and mean per-simulation time U-1,…,U-4

Figure III.9 presents the total time complexity for solving one pattern: ->O($n^{4,7}$). This is quite high complexity.

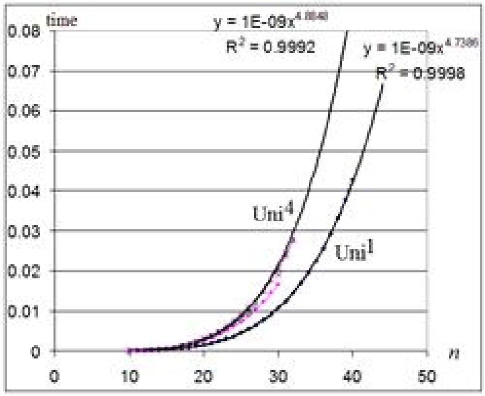


Fig.III.9. Time complexity of the LPF-algorithm for Uni-1, Uni-4 traffic

3.5 Conclusions

The computer simulation of the PIM algorithm is based on its generalized network model. The GNs-model presents a clear parallelism of the algorithm processes. The formal apparatus used makes it possible to obtain quantitative characteristics as a result of the operation of a modeling algorithm.

A family of patterns has been developed to simulate irregular traffic. The results of the computer simulations carried out on the grid clusters of CERN and IICT-BAS are shown. The simulations use the PIM conflict-free scheduling algorithm specified by the Generalized Networks apparatus. In the design of the developed GNs-model, the additional position p is introduced, which allows to obtain symmetry in the specification of the three phases of the algorithm.

The Generalized Networks apparatus is applied to describe the LPF-algorithm for conflict-free scheduling modeling. The GNs-model leads to conflict-free scheduling. It is shown that the execution time of a simulation depends on the degree of the dimensionality n -> $O(n^{4,7})$.

.

As a result, the following conclusions have been drawn:

- It is shown that the apparatus of generalized networks can be used to model and analyze parallel processes in algorithms used in switches.
- • GNs-models of PIM and LPF algorithms are proposed. Modeling in the context of GNS leads to the explicit designation of the synchronization points of parallel processes in the PIM algorithm.
- The simulation of the PIM algorithm for the proposed *Chao*$_i$ pattern family shows that the throughput tends to an upper bound that is less than 100% and greater than 76%.

Chapter 4 - A new MiMa conflict-free scheduling algorithm.

The Generalized Nets apparatus is used in this chapter to describe our MiMa (Minimum of Maxima) algorithm, which computes the conflict-free scheduling of a matrix switch packet switch with Virtual Output Queues (VOQ).

4.1 Conflict criteria for "weight" algorithms.

We will define 3 forms of the conflict criterion of the elements of the virtual output queues of the inbound traffic.

Weak form: The elimination of conflicts starts with those located in only the column (or only the row) of the query matrix ***T*** where there is a maximum number of conflicts.

Strong form: The elimination of conflicts starts with those located in the column, and then in the row, of the query matrix ***T*** where there is a maximum number of conflicts.

Complete form: The elimination of conflicts starts with those located in the matrix ***T*** element that has the maximum sum of conflicts in its rows and columns.

For the "weak" and strong "form" we introduce the terms "row conflict weight" *Col(n)* and "column conflict weight" *Row(n)* . For the "full" form of the criterion - the term "request weight $w_{i,j}(n)$, for a request from input i to output j" is defined in [40]. We propose the MiMa algorithm implementing the strong form. The goal is to get more speed while sacrificing a bit of THR.

4.1.1 The MiMa algorithm

We will give a brief description of the MiMa algorithm (for the purpose of calculating the matrix Q_1).

Start. First we enter n and ***R*** (n, r_{ij}, $i,j \in \{1,\ldots,n\}$).(***R*** is a copy of ***T***).

(1) The column vector *Col(n)* is calculated, which consists of the sum of the conflicts of each row (weight of the conflicts in the row). If there are no requests (the column vector contains only 0-elements)) **then** we go to **End**; **otherwise** we continue.

(2) The row vector *Row(n)* which consists of the sum of the conflicts in each column (the weight of the conflicts in the column), is also calculated. In the row vector, we select the largest element that defines the column with the most conflicts. In the column vector, we select the largest element that defines the input with the most conflicts.

(3) If there is a query at the intersection of the row and column, then we take that query as an element of the conflict-free matrix Q_1. We temporarily record a zero weight for these input and output rows. We go to (**1**); **otherwise** (if there is no query) we select an element of the vector *Col(n)* that is closest to the maximum weight value (row selection remains the same).

(4) We check if there is an element (query) at the intersection, and then proceed as in **(3)** (omitting the details). As a result, in the selected column of ***R***, we have a query selected for commutation (if such a query exists at all). The row and column containing the selected request are excluded from the calculations on Q_1.We go to **(1)**.

End.

The next elements of Q_1 are calculated by repeating the **(1)-(4)** procedure. As a result, the matrix Q_1 may consist of the elements with maximum conflict weights from $\boldsymbol{R}$. The final matrix Q_m will contain only conflict-free queries.

4.2 Constuct of a Generalized Nets model of a new "weight" algorithm MiMa.

The information processing steps in the MiMa-algorithm are clearly defined and therefore we can effectively describe these processes using Generalized Networks. We build an OM-model based on the following rules:

- to each comparison operator (from the algorithm) corresponds a single transition in the OM-model;

- to each group of successive assignment operators (from the algorithm) corresponds one transition in the OM-model.

The formal description with OM is given in the full text.

The model of the algorithm is for a node with c *n* inputs and *n* outputs. The graphical representation is shown in Figure IV.1. At the first instant of the current modeling time, the core (one) enters position l_1 (start). The end of the execution of the MiMa-algorithm is indicated by the arrival of one core in position l_{22} (end). At this point, position l_{20} contains the core of the final conflict-free schedule (the core represent the solutions $Q_1, Q_2, \ldots, Q_k$).

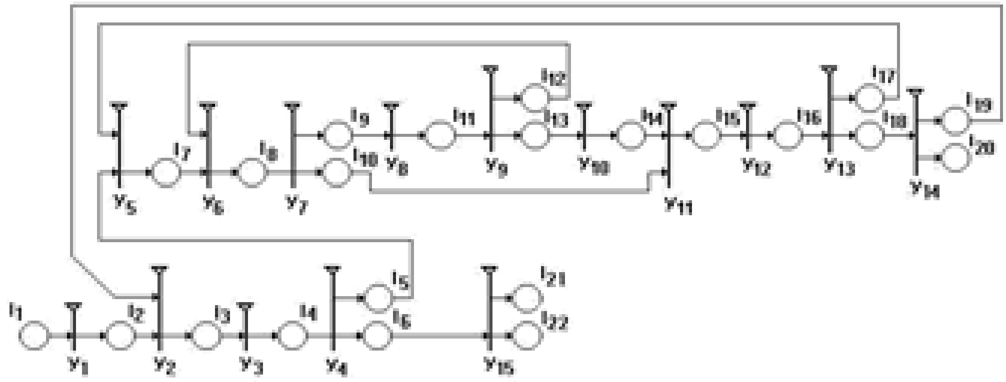


Fig.IV.1. Graphical representation of the OM-model of the MiMa-algorithm.

4.2.1 Formal description of MiMa algorithm

The form of the first transition in the GNs-model is::

$Y_1 = \langle L_1', L_1'', r_1, \vee(L_1') \rangle$ where $L_1' = \{ l_1 \}$;$L_1'' = \{ l_2 \}$

and the index matrix is: $r_1 = \begin{array}{c|c} & l_2 \\ \hline l_1 & true \end{array}$

The characteristic function is:

- Φ_2 =" k=0, Q_k=0, *Col(n)*, *Row(n)* "

  :

  Y_2=< L_2', L_2'', r_2, $\vee$ (L_2') > where L_2' ={l_2, l_{19}}; L_2'' ={ l_3}

$$r_2 = \begin{array}{c|c} & l_3 \\ \hline l_2 & true \\ l_{19} & true \end{array}$$

- Φ_3 =" *Col_m(n)* :=*Col(n)*; *Col_t(n)* :=*Col_m(n)*; *Row_t(n)* := *Row(n)* "

  Y_3=< L_3', L_3'', r_3, $\vee$ (L_3') > where L_3' ={ l_3};L_3'' ={l_4}

$$r_3 = \begin{array}{c|c} & l_4 \\ \hline l_3 & true \end{array}$$

- Φ_4 =" $Y\_0 := \sum_j Row(j)$ ", (j=1,2,…,n) .

  Y_4=< L_4', L_4'', r_4, $\vee$ (L_4') > where L_4' ={l_4};L_4'' ={l_5 , l_6}

$$r_4 = \begin{array}{c|cc} & l_5 & l_6 \\ \hline l_4 & \neg end & end \end{array}$$

The predicate in r_4 has the following form:

- end= " $Y\_0 = 0$ "

The characteristic functions are:

- Φ_5="r :=0; k := k+1"
- Φ_6 =" to end "

  Y_5=< L_5', L_5'', r_5, $\vee$ (L_5') > where L_5' ={l_5, l_{17}};L_5'' ={ l_7}

$$r_5 = \begin{array}{c|c} & l_7 \\ \hline l_5 & true \\ l_{17} & true \end{array}$$

- Φ_7 =" j_num := max[*Row_t(n)]*"

  Y_6=< L_6', L_6'', r_6, $\vee$ (L_6') > where L_6' ={l_7, l_{12}};L_6'' ={ l_8}

$$r_6 = \begin{array}{c|c} & l_8 \\ \hline l_7 & true \\ l_{12} & true \end{array}$$

- Φ_8 =" $\boldsymbol{X} := \sum_i Col\_t(i)$ ", (i=1,2,…,n) .

  Y_7=< L_7', L_7'', r_7, $\vee$ (L_7') > where L_7' ={l_8};L_7'' ={l_9 , l_{10}}

$$r_7 = \begin{array}{c|cc} & l_9 & l_{10} \\ \hline l_8 & \neg sum\_X & sum\_X \end{array}$$

The predicate in r_7 has the following form:

- sum_X= " $X = 0$ "

The characteristic functions are:

- Φ_9 =" *"
- Φ_{10} =" * "

Y_8=< L_8', L_8'', r_8, $\vee$ (L_8') > where $L_8' = \{l_9\}$;$L_8'' = \{l_{11}\}$

$$r_8 = \begin{array}{c|c} & l_{11} \\ \hline l_9 & true \end{array}$$

- Φ_{11} =" i_num := max[*Col_t(n)*] " .

Y_9=< L_9', L_9'', r_9, $\vee$ (L_9') > where $L_9' = \{l_{11}\}$;$L_9'' = \{l_{12}, l_{13}\}$

$$r_9 = \begin{array}{c|cc} & l_{12} & l_{13} \\ \hline l_{11} & \neg match & match \end{array}$$

The predicate in r_9 has the following form:

- match= " $R(i\_num, j\_num) \neq 0$ "

The characteristic functions are:

- Φ_{12}="*Col_t(i_num)* := 0"
- Φ_{13} =" r := r+1 "

Y_{10}=< L_{10}', L_{10}'', r_{10}, $\vee$ (L_{10}') > where $L_{10}' = \{l_{13}\}$;$L_{10}'' = \{l_{14}\}$

$$r_{10} = \begin{array}{c|c} & l_{14} \\ \hline l_{13} & true \end{array}$$

- Φ_{14} =" $Q_k(i\_num, j\_num):=1$; $R(i\_num, j\_num):=0$;$Col\_m(i\_num):=0$; $Row(j\_num):=Row(j\_num)-1$; $Col(i\_num):=Col(i\_num)-1$ " .

Y_{11}=< L_{11}', L_{11}'', r_{11}, $\vee$ (L_{11}') > where $L_{11}' = \{l_{10}, l_{14}\}$;$L_{11}'' = \{l_{15}\}$

$$r_{11} = \begin{array}{c|c} & l_{15} \\ \hline l_{10} & true \\ l_{14} & true \end{array}$$

- Φ_{15} =" $Col\_t((n):=Col\_m(n)$; $Row\_t(j\_num):=0$"

Y_{12}=< L_{12}', L_{12}'', r_{12}, $\vee$ (L_{12}') > where $L_{12}' = \{l_{15}\}$;$L_{12}'' = \{l_{16}\}$

$$r_{12} = \begin{array}{c|c} & l_{16} \\ \hline l_{15} & true \end{array}$$

- Φ_{16} =" $Y := \sum_j Row\_t(j)$ ", (j=1,2,…,n) .

Y_{13}=< L_{13}', L_{13}'', r_{13}, $\vee$ (L_{13}') > where L_{13}' ={l_{16}};L_{13}'' ={l_{17} , l_{18}}

$$r_{13} = \begin{array}{c|cc} & l_{17} & l_{18} \\ \hline l_{16} & \neg sum\_Y & sum\_Y \end{array}$$

The predicate in r_{13} has the following form:

- sum_Y= " $Y = 0$ "

The characteristic functions are:

- Φ_{17}=" * "
- Φ_{18} =" $Q_k$isready "

Y_{14}=< L_{14}', L_{14}'', r_{14}, $\vee$ (L_{14}') > where o L_{14}' ={l_{18}};L_{14}'' ={l_{19} , l_{20}}

$$r_{14} = \begin{array}{c|cc} & l_{19} & l_{20} \\ \hline l_{18} & true & true \end{array}$$

The characteristic functions are:

- Φ_{19}=" * "
- Φ_{20} =" Q_k , k, r "

The form of the last transition in GNs model is:

Y_{15}= <L_{15}', L_{15}'', r_{15},$\vee$ {L_{15}'}> where L_{15}'={ l_6};L_{15}'' ={ l_{21}, l_{22} }

$$r_{15} = \begin{array}{c|cc} & l_{21} & l_{22} \\ \hline l_6 & error & \neg error \end{array}$$

The predicate in r_{15} has the following form:

- error= " $k = 0$ "

The characteristic functions are:

- Φ_{21} =" input error "
- Φ_{22} =" stop "

Each of the transitions has the same priority. This also applies to cores. The model provides information on the number of switching configurations of the switching field (the variable *k*) as well as the number of packets transmitted during one switching (the variable *r*). This information will be used to calculate the average THR. Other information may also be collected.

4.2.2 Results of the grid simulations

In this study, we use extended and mirror patterns. The example for the case (for size 3x3) is shown in Figure IV.2. In this case, the resulting throughput is averaged over *n* simulations for each switch size (*n x n).*

$$R^{1}_{(3\times 3)_z} = \begin{bmatrix} 1 & 1 & 2 \\ 1 & 2 & 1 \\ 2 & 1 & 1 \end{bmatrix} \Rightarrow \begin{bmatrix} 1 & 1 & 2 \\ 1 & 2 & 1 \\ 2 & 1 & 1 \end{bmatrix}, \begin{bmatrix} 1 & 2 & 1 \\ 2 & 1 & 1 \\ 1 & 1 & 2 \end{bmatrix}, \begin{bmatrix} 2 & 1 & 1 \\ 1 & 1 & 2 \\ 1 & 2 & 1 \end{bmatrix}$$

Fig.IV.2. Example of matrices of *Chao*^1_z types for size (3x3)

The transition from the OM-model to the calculation program is carried out as shown in our publication. The code was created using the Vfort package (open source) [46]. It was compiled on the grid-cluster BG01-IPP of IICT-BAS (www.grid.bas.bg). The resulting code is executed locally on the grid cluster. The operating system is ScientificLinux rel.6.5. Resources: up to 16 processors (2 blades), 32 threads, 2GB RAM. Execution time is < 72 hours.

In the figures below, *Chao*^i_z is denoted as Cz-i for i=1,2,… Figure IV.3 shows the computer simulation results of the MiMa-algorithm with input parameter *Chao*^1_z. The throughput for one simulation is calculated as $(2i(n-1))/k$, i.e. normalized to the full throughput of the switching node (1 equals 100%); time is given in seconds.

Figure IV.4 presents the results with *Chao*$^{10}_z$ as an input parameter, and Figure IV.3 presents the results with *Chao*$^{100}_z$ as input parameter. Comparing the two figures, we see that the throughput increases when the number of patterns (i) is large, and the execution time also increases. Results show that the rate at which a certain limit is approached also increases (Figure IV.5).

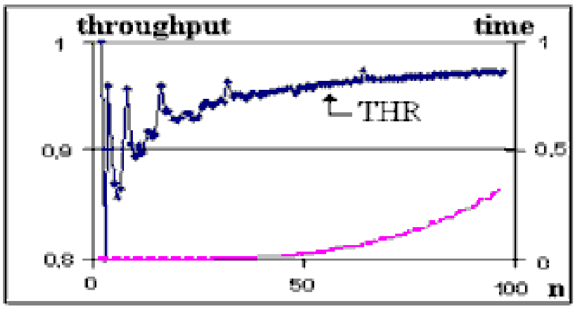


Fig.IV.3 THR and time at *Cz-1*

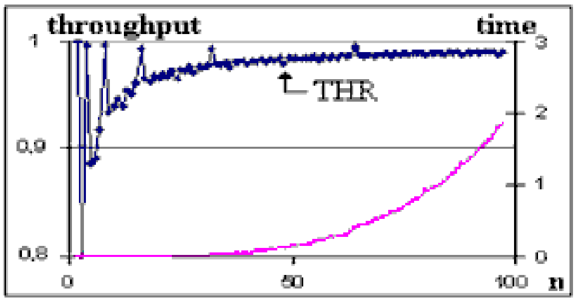


Fig.IV.4. THR and time at *Cz-10*

Figure IV.6 presents the comparison of execution time for *Chao*$^{10}_z$ and *Chao*$^{100}_z$ patterns. Here y is the approximation of the polynomials for the size of the matrix ***R*** from 13x13 to 97x97; R2 is the correlation in the data. Figures 3 to 5 show that the THR limit is less than 100%, but tends towards this. It is also shown that the execution time increases linearly with the increase of the pattern index i for i=1,10,100. This corresponds to a linear increase in the size of the input buffer.

Figures 3 to 5 show that the THR limit is less than 100%, but tends towards this value. It is also shown that the execution time increases linearly with the increase of the pattern index i for i=1,10,100. This corresponds to a linear increase in the size of the input buffer.

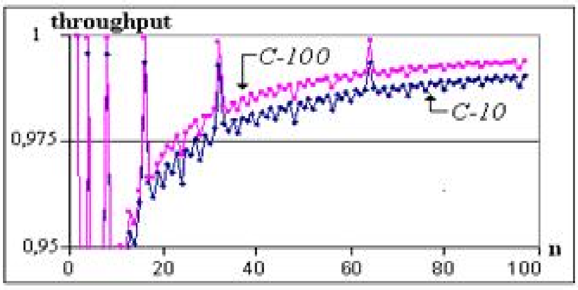


Fig.IV.5. Comparison of THR *Cz-10, Cz-100*

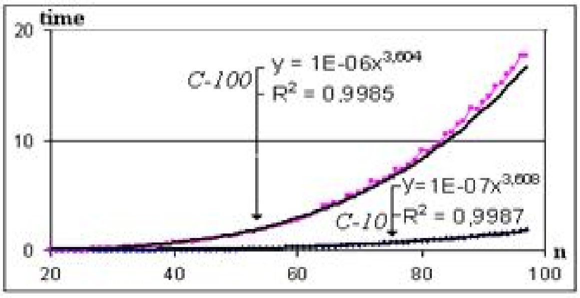


Fig.IV.6 Time for *Cz-10, Cz-100*

The solution time of for the entire matrix ***T*** is proportional to O($n^{3,6}$). The time includes the computation of i.2.(*n*-1) commutation solutions. So the power of the time complexity for one solution (one matrix $\boldsymbol{Q_i}$) should be reduced by one. Therefore, for a Chao$^{i}_{z}$ pattern, the execution time of the MiMa-algorithm is proportional to the power of n equal to 2.6. This is very close to the known minimum, theoretically calculated at O(N5/2) [40] for the complexity of weight algorithms such as the MiMa-algorithm.

4.3 Verification of the "always choose the heaviest coefficient" principle.

The new algorithm MiMa (Minimum of Maxima) is proposed by the author of this thesis. The throughput of the switch in MiMa algorithm runtimes approaches 100% for both uniform Uni traffic and "hot spot" (Chao) traffic. However, in both cases there are observed certain fluctuations of the THR as it approaches the limit, in contrast to the monotonous approach to the THR limit for the classical PIM algorithm. The question arises whether it is possible to modify the MiMa-algorithm to obtain a "smoother" course of its THR. And maybe to increase it.

For this purpose, a version of the MiMa algorithm with a new initial element selection (min-max) compared to the original version of the algorithm (max-max) was investigated.

4.3.1. Selecting an initial element for conflict-free switching in MiMa

The essence of the algorithm is given by the characteristic function Φ_7 of the transition Y_5 and by the characteristic function Φ_{11} of the transition Y_8.

The MiMa-algorithm implements the "max-max" type selection - the element with the maximum "conflict weights" on output and input (in this order) is selected. Therefore, 3 modifications of the algorithm are possible: selection of the "min-min" type; type selection "max-min"; "min-max" type selection. The THR simulation result of the first modification is not good. Here we will show the results for the "min-max" option.

4.3.2 Computer simulation of the THR

Inbound traffic uses the Uni model of uniformly distributed traffic with uniform load. Figure IV.7 shows the results of the algorithm for the pattern families U-1 and U-10. The designation U-i is used for traffic type T^i. The dimension n of the input matrix $T^i(_{kxk})$ is shown along the horizontal coordinate axis. The left vertical axis shows the throughput of the algorithm. On the right vertical axis is shown the time to calculate the graph (in seconds). When comparing the THR of the algorithm for the family of patterns U-1,10,100, it increases with the increase of the index i and tends to the limit THR (100%). The computation time is of the order of the third power of n.

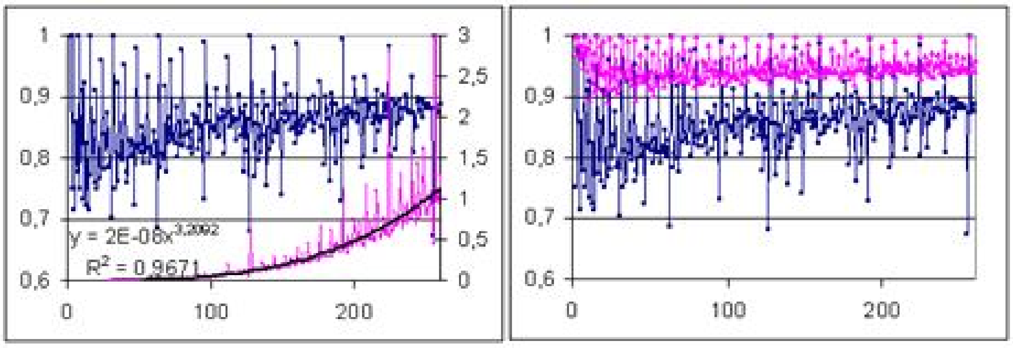


Fig.IV.7 THR for modified $MiMa_m$ (U-1, time) and $MiMa_m$ (U-1, U-10).

4.3.3. Comparison of the type selection options

Compared to the min-min case, the obtained result is undoubtedly "better", so min-min is not shown here. But compared to the max-min modification, the THR result was unexpected for us - the THR is the same. So we had to compare the computation time. This is shown in the full text. The last studied modification has a slightly better calculation time (less time spent).

The main finding is that changing the selection of an initial switching element did not result in a “smoothing” or increase in THR when approaching the limit. In addition, the rate of increase of THR to the limit slows down (relative to the max-max case). The simulation results lead to the conclusion that the max-max variant gives the best result.

4.4 Conclusions

Our simulation results of the MiMa-algorithm show that the throughput tends to the upper bound, which has a maximum value of 100%. We have shown that the complexity of the MiMa-algorithm tends to the optimal theoretical one. Therefore, MiMa can serve as a benchmark for the speed of "weighted" type algorithms.

A version of the MiMa algorithm with a new initial element selection (min-max) compared to the original version of the algorithm (max-max) was investigated. The obtained results are compared with the simulation results of two other choices (max-min) and (min-min) for the initial element. It is concluded that the application of the "always choose the heaviest coefficient" principle (the original MiMa algorithm) gives the best results for node throughput.

Chapter 5 - Numerical procedure for an exact upper bound on the throughput of algorithms

This chapter describes a numerical procedure for calculating the upper limit of the PS. If it exists, then the solution is unique. In this procedure, we use the results of the computer simulation of THR, conducted on the grid-structure BG01-IPP of IIKT-BAS (www.grid.bas.bg) and the supercomputer "AVITOHOL" of BAS. Our PS modeling starts with the PIM-algorithm, the Chaoi model for “hotspot” (Chaoi) incoming traffic [36] and 100% load intensity at each entrance. The obtained results give the upper limit of PS for n ∈ [3, 100], which allows us to calculate the limit of THR for n->infinity: the result is 0.775+/-0.001

5.1 Input Data. Setting the task. Existence of the solution.

Here we use a Chao pattern matrix family for input data.. If we consider the Chao pattern matrices as an additive sum between Uni-traffic and non-conflict traffic, we have an upper bound of PS=1-1/(2e-1) = 77,460 .. %=.

Figure 1 shows the results of the algorithm. The abscissa shows the dimension n of the matrix T (from 2 to 70). The ordinate in Fig.1 shows PS. Each point is the mean value for 10,000 simulations.

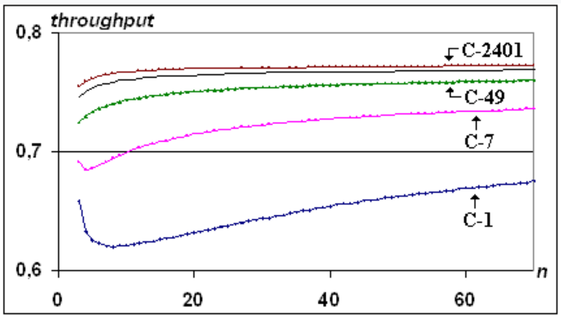


Fig.V.1 THR of the PIM-algorithm for Chao-traffic.

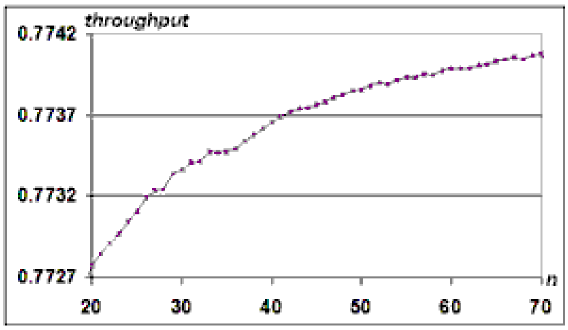
The data in Figure V.1 show an increase in throughput as n increases for each template used. When "switching" to a "heavier" pattern (increasing i), the minimum shifts to the left, and the rate of increase of PS decreases. This gives us reason to assume the existence of an upper limit.

5.2 Procedure for calculating the upper limit of the PS in a determined operating range.

The described procedure calculates the convergence of the THR of the commutator to a specific value (upper limit) for each resistance of n according to the result of the computational experiments. An image is given in the main text and this is given by Fig.1 – for the selected pattern sequence Chaoi , i = 1,m1,m2,m3,m4, where m=7 (Chao1, Chao7, Chao49, Chao343, Chao2401,) . First of all, it is important for us to check the theorem δ=m-½ (this is the heuristic solution, for m=2,3,4,5 it is true).

The modeling results show that excluding the PS minimum displacement zone (2 < n < 20), the PS differences between the "steps" of the templates decrease in geometric progression. It is known from the theory that an infinite numerical series of the form $1/a + 1/a^2 + 1/a^3 +\ldots+1/a^i+\ldots$, when a>1, converges for $i\to\infty$ to the value Sum(a)=1/ (a-1). Assuming that in our case the differences Δ2, Δ3,... form such a series with a=2.64575, then the convergence coefficient for the displayed results is Sum(2.64575)=1/(2.64575-1)= 0.607625.

Then we obtain the values of the upper bound for specific n using the template with the largest number - Chao2401, as follows: LimitPS(n) = PS(Chao2401 (n)) + Sum (2.64575). Δ4 (n)= PS(Chao2401 (n)) + 0.607625. Δ4 (n) .(in figure V.2).



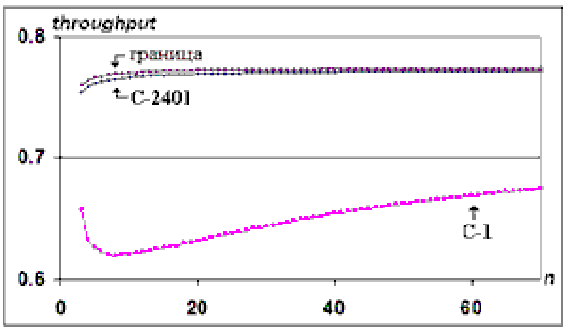


Fig.V.2 The calculated upper limit of THR. Fig.3 The initial data and the border of the THR.

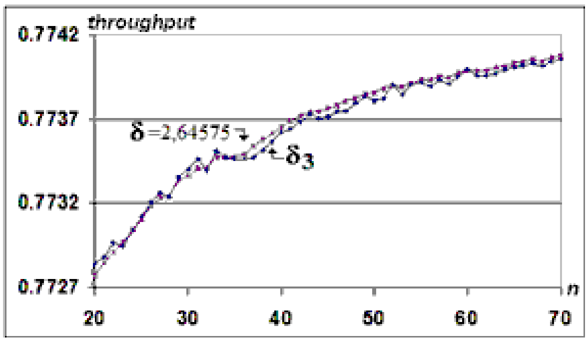
The largest values are THR (Chao2401(70))=0.772005, Δ4(70)=0.003401, Limit Efficiency(70)=0.774031. Relative to the data for the first and largest template used in the simulations, the PS boundary is shown in Figure V.3.

5.3 Accuracy problems in operating range.

Consider the case where the upper limit is calculated not by multiplying the difference Δ_4 by the constant Sum(2.64575), but by multiplying by a variable. Namely $\mathrm{Sum}(\delta_3)=1/(\delta_3-1)$. Thus, we can estimate the influence of the inaccuracy of the simulation data on the final result (upper bound).

The comparison between the curve of Fig.V.2 and the new result is shown in Fig.V.4.

The results already obtained can be compared. We will do this with the calculated upper bound for "step" m=2. There, $\delta_5\approx1.41\pm 0.025$ was chosen as the best result.



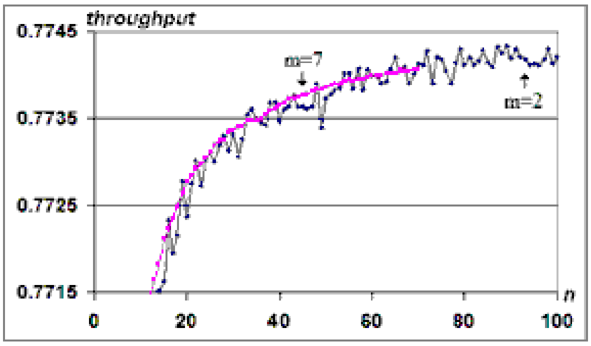


Fig.V.4 Upper limit of THR at δ_3=const, δ_3=var Fig.V.5Upper limit of THRatm=7,m=2.

Accordingly, the upper bound was calculated on the data from Chao64 and Chao32 templates (10,000 simulations). A comparison of this bound with the one obtained above is shown in Figure V.5. Specifically, we can derive the following benefit. The estimate for the limit value of PS (based on the data for m=2) is 0.775± 0.001. From Figure V.8 we conclude that this is a slightly inflated estimate. We believe that the limit value of the PS has a value of 0.7748± 0.0005 (at $n \to \infty$ theoretically 1-1/(2e-1)).

5.4 Specification of the numerical procedure.

In our research, we solve the problem in two steps: proving that the solution exists; calculating the solution. The formal description of the procedure is given in the full text. If there is an upper bound on the throughput of a switch, it is unique.

The next research question is whether the procedure is robust. The case of small, asymptotically decreasing perturbations of incoming traffic is considered.

5.5 Computer simulations to study the stability of the numerical procedure

We introduce perturbations to the Chao model as follows: we first modify the template family to the “mirror” variant (described in the full text). Second, we reduce the number of requests in the selected (first) input row by one (minus one request).

Our computer simulations confirm the applicability of the proposed procedure with modified traffic load patterns. The result of the simulations shows (in the main text) that the numerical procedure is robust in the sense that small values of the intensity perturbations of the input requests lead to small changes in the output. The obtained results give an upper limit of PS for $n \in [3, 97]$, which allows us to estimate the limit of PS (THR) of MiMa-algorithm for n>infinity. The obtained value is 100%

5.6 Exact upper limit in a specified working interval

THR of PIM-algorithm with Chao-input

The THR upper bound result for PIM-algorithm with Chao-input (m=2) is shown in Figure V.6.

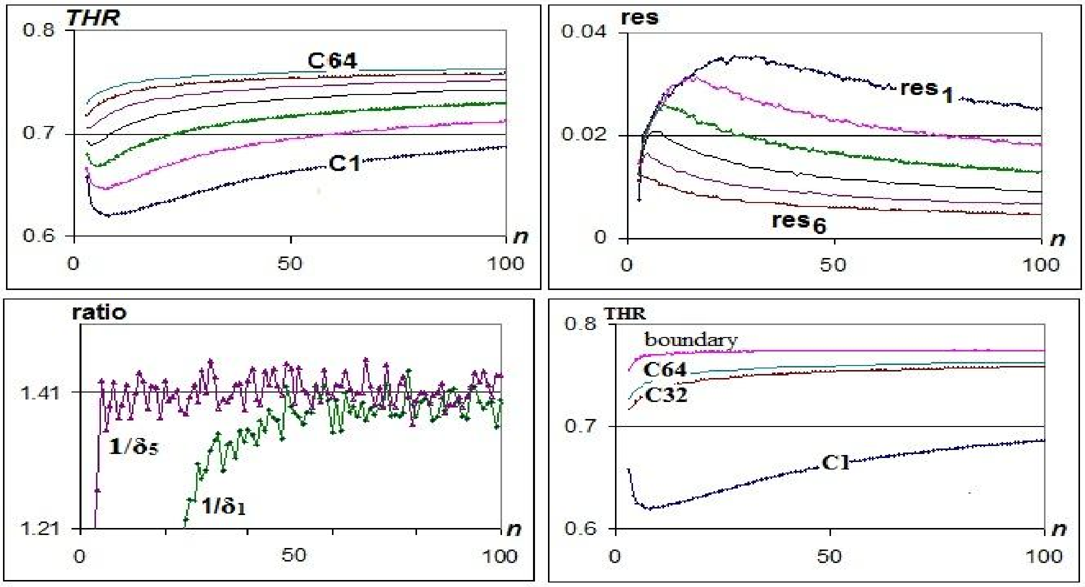


Fig.V.6 Upper limit of PS (THR) at δ_5=const=1.4142 and m=2.(PIM, Chao)

The ratio δ with increasing pattern index quickly reaches the expected value 1/1.41 (m=2). This is a consequence of the property of the PIM-algorithm to process the inputs equally. At the end of the operating range (n=100) we have the largest value for THR.

THR of LPF- algorithm with c Uni-input

Figure V.7 shows the upper bound result of THR for LPF-algorithm with Uni-input (m=2). The ratio δ with increasing pattern index tends to the expected value 1/1.41 (m=2). Achieving and is expected at larger pattern indices. At index i= 131072, a value of 1/1.46 is obtained at the end of the working range. This is clearly a consequence of the inherent properties of the LPF-algorithm - we assume due to the presence of "starvation" - unequal processing of the inputs. Therefore, the calculated upper limit of THR is inflated. At the end of the operating range (n=100) we also have the largest value for THR.

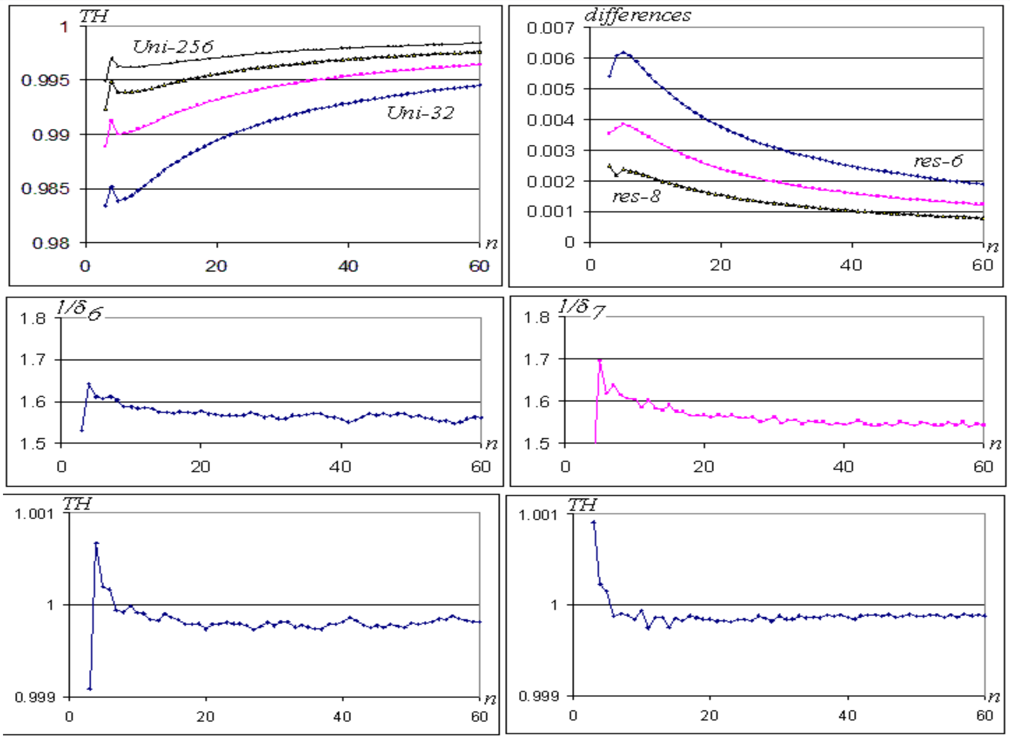


Fig. V.7 Upper limit of PS (TH) at δ6=var and m=2. (LPF, Uni)

THR of MiMa-algorithm, Chao-input with decaying perturbation

Figure V.8 shows the result for the upper limit of PS for LPF-algorithm with Chao-input (m=2) under decaying input disturbance. The ratio δ with increasing pattern index predictably tends to the expected value 1/1.41 (m=2). Achieving the upper bound is also expected at larger pattern indices.

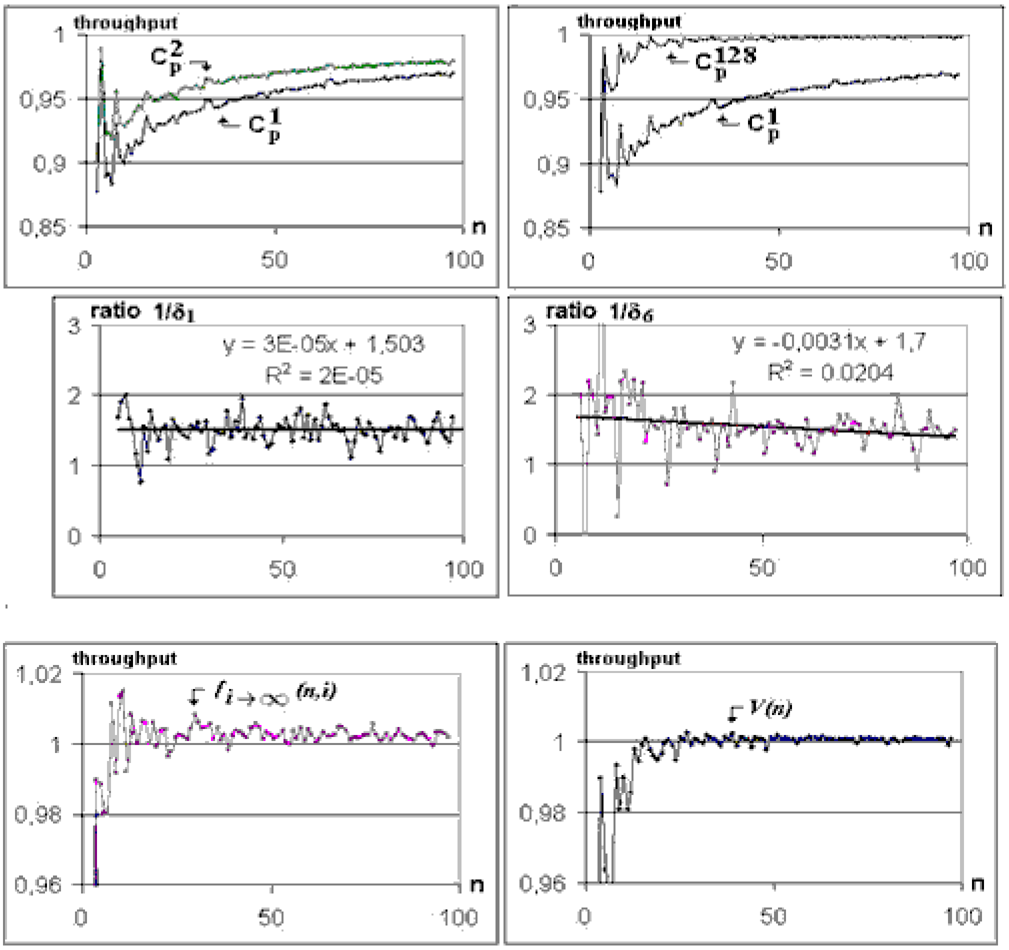


Fig. V.8 Upper limit of PS at δ6=var and m=2. (MiMa, Chao damping disturbance)

We also assume here non-equivalent processing of inputs because of the presence of "starvation" (MiMa uses the same kind of criterion as LPF, but its weaker form). Therefore, the estimated upper limit of PS is more inflated than with LPF, because MiMa generally lags behind LPF in THR.

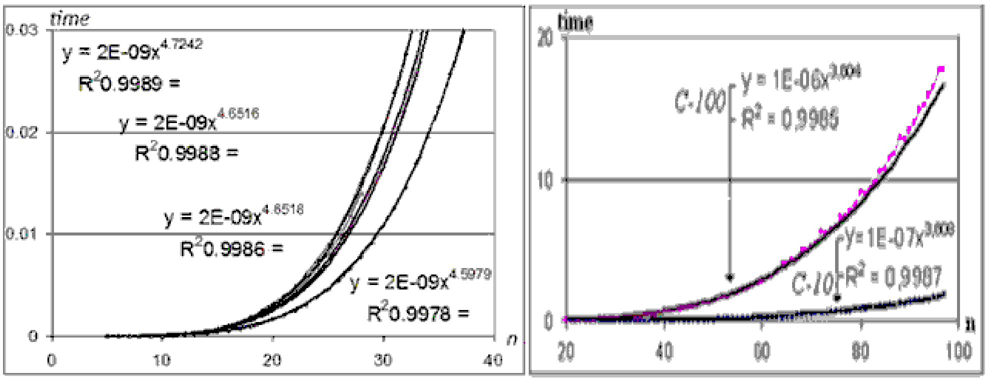


Fig.V.9 Solution time of the entire input matrix (LPF-left, MiMa, - right)

We can make time comparisons for the solution of the entire input matrix in Figure V.9. The time complexity is the estimate for one commutation. Therefore, since the solutions are respectively κ_{sol}, the degree of time complexity for one solution (commutation Q) will have to be reduced. More precisely, for the corresponding values it turns out (in the main text) that we have (O($n^{2.9}$)) for LPF, and (O($n^{2.6}$)) for MiMa.

Conclusions that can be drawn:

The results of our simulation of the algorithm show that the MiMa throughput tends to the upper bound, which has a value of 100%. In the operating range up to n=100, Mima lags behind PS by 3-2%, and the difference is decreasing. The complexity of the MiMa-algorithm (O($n^{2.6}$)) is very close to the optimal theoretical one. (O($n^{2.5}$)). Therefore, a further increase in the execution speed can be achieved by using parallel computations of the algorithm's transitions and also by modifications of the deterministic discipline for selecting a principal for the conflict-free solution. Can the THR be "smoothed out" at the same time - this is a question for future research

Conclusion

The dissertation presents studies of the modeling of the switching processes in a communication node using the Generalized Networks (GNET) apparatus. Formal GN-models of 4 known algorithms for calculating a conflict-free schedule in a packet switch with a matrix switch (crossbar switch node) are specified. A new algorithm for conflict-free scheduling (MiMa-algorithm) is proposed. The model of this algorithm is presented formally in the form of a Generalized Network. A comparison of its performance with that of other algorithms is discussed. The loss of 2 to 3 % of the bandwidth is the price for its speed - the MiMa algorithm is at the lower limit of time complexity (time forinplementation) for its class ("weighted") algorithms. The results were obtained in large-scale computer simulations of the grid-structure of IIKT-BAS and of the supercomputer "Avitochol" of BAS, by applying a newly developed numerical procedure, resistant to asymptotically damping disturbances, to the numerical data from the simulations. possible adequate and unambiguous comparison of the throughput of conflict-free scheduling algorithms in a given operating range.

Summary of received results

In view of the work carried out in the dissertation and the conclusions obtained in the course of the research and presented above, the following scientific and application results can be formulated:

1 .A new MiMa (MinimumofMaxima) algorithm for conflict-free scheduling in a matrix switch packet switch is synthesized and investigated. The throughput of the algorithm approaches 100%, and its execution time complexity is ($O(n^{2.6})$). The theoretical limit for the class of "weighted" algorithms to which MiMa belongs is ($O(n^{2.5})$).

2.GN-models of 4 classical algorithms for conflict-free scheduling in a matrix switch packet switch are modeled with the Generalized Networks (GNs) apparatus and investigated.

3. Four pattern families for 4 classical (uniform, Chang, Chao, Rojas-Chessa) types (i.i.d. Bernoulli) incoming traffic are synthesized, designed for large-scale computer simulations of the throughput (PS) of conflict-free scheduling algorithms, at 100 % load on incoming lines.

4. A numerical procedure is developed to calculate an exact upper bound on throughput (UP) of contention-free scheduling algorithms in a matrix switch packet switch. The limit is calculated for a given operating range of the switching field (n) in large-scale computer simulations of the THR with the synthetic patterns of incoming traffic. The procedure is robust to asymptotically decaying perturbations. The procedure is applied to the results of computer simulations of the PS THR of the synthesized GN-models.

Future development

The main directions for future research on the topic of the dissertation include:

Research the possibilities of improving the characteristics of the proposed MiMa-algorithm, by modifying the discipline for selecting the main element, similar to those used here when modifying the models of the "Wave Front" and "Monitoring" algorithms.

Investigating the possibilities of obtaining a load of less than 100% on the incoming lines by modifying the families of incoming traffic templates proposed here.

Investigating the limits of validity of the proposed numerical procedure for determining an exact upper limit of throughput, for non-i.i.d. Bernoulli types of incoming traffic (such as burst-growing and self-modifying).

Publications

1. **Tashev, T.D.,** Marinov, M.B., Arnaudov, D.D., Monov, V.V. Computer Simulations for Determining of the Upper Bound of Throughput of LPF-Algorithm for Crossbar Switch. AIP Conference Proceedings, 2505, American Institute of PhysicsInc., NY 11747-4501, USA, 2022, ISBN: 978-073544396-9, ISSN: 0094243X, DOI:10.1063/5.0103594, 080030**. SJR (Scopus):0.19**
2. **Tashev, T.** A generalized network model of the wavefront algorithm for a matrix switch packet switch. Collection of reports from the Annual University Scientific Conference 2022, Veliko Tarnovo, Bulgaria, 2022, volume 6, Publishing Complex of NSU "Vasil Levski", 2022, ISSN:1314-1937, pp. 217-225 (in bulgarian)
3. **Tashev, T. D.,** Marinov, M. B., Tasheva, R. P., Alexandrov, A. K.. Generalized nets model of the LPF-algorithm of the crossbar switch node for determining LPF-execution time complexity. AIP Conference Proceedings, 1, 2333, American Institute of Physics Inc., NY 11747-4501, USA, 2021, ISBN: 978-073544077-7, ISSN: 0094243X, DOI:10.1063/5.0042856, 090039. **SJR (Scopus):0.177**
4. **Tashev T.,** Monov V., Petrov P. Efficiency of the “choose the maximum weight” principle for calculating the throughput of a packet switch using the MiMa algorithm. Proceedings of the XXI International Scientific Conference DCCN-2018, Peoples' Friendship University of Russia, Moscow, Russia, 2018, ISBN: 978-5-209-09082-3, p.71-78 **(eLibrary)** (in russian)
5. **Tashev T.,** Monov V., Tasheva R.. High Performance Computations for Study the Stabilityof a Numerical Procedure for Crossbar Switch Node. In: Dimov I., Faragó I., Vulkov L. (eds) Numerical Analysis and Its Applications. NAA 2016., LNCS, volume 10187, Springer, Cham, 2017, ISBN: 978-3-319-57098-3, ISSN: 03029743, DOI:10.1007/978-3-319-57099-0_76, p.665-673. **SJR (Scopus):0.28, JCR-IF (Webof Science):0.302**
6. **Tashev T.,** Marinov M., Monov V., Tasheva R.. Modeling of the MiMa-algorithm for crossbar switch by means of Generalized Nets. Proceedings ot the 2016 IEEE 8th International Conference on Intelligent Systems (IS), 4-6 Sept. 2016, Sofia, Bulgaria., IEEE, 2016, ISBN: 978-1-5090-1354-8, DOI: 10.1109/IS.2016.7737486, p.593-598 **(WoS, Scopus)**
7. **Tashev T.,**Monov V.. Modeling of the hotspot load traffic for crossbar switch node by means of Generalized Nets. 6th IEEE International Conference on Intelligent Systems, IEEE, 2012, ISBN: 978-146732782-4, DOI:10.1109/IS.2012.6335214, p.187-191 (**Scopus)**

Citations

I. Tashev T., Monov V.. Modeling of the hotspot load traffic for crossbar swit chnode by means of Generalized Nets. 6th IEEE International Conference on Intelligent Systems, IEEE, 2012, ISBN: 978-146732782-4, DOI:10.1109/IS.2012.6335214, pp.187-191, **2012**

Cited in::

1. Balabanov, T. , Zankinski, I. , Barova, M. Strategy for individuals distribution by incident nodes participation in star topology of distributed evolutionary algorithms. J. Cybernetics and Information Technologies, Volume 16, Issue 1, 2016, Pages 80-88. DOI: 10.1515/cait-2016-0006 (WoS), **2016**
2. Gocheva, P. V., Hinov, N. L., Gochev, V. P. "Modeling of Buck DC-to-DC Converter with Generalized Nets". Proceedings of the 2018 IEEE XXVII International Scientific Conference Electronics - ET"2018. Sozopol, Bulgaria. DOI: 10.1109/et.2018.8549605, **2018**

II. Tashev T., Marinov M., Monov V., Tasheva R.. Modeling of the MiMa-algorithm for crossbar switch by means of Generalized Nets. Proceedings ot the 2016 IEEE 8th International Conference on Intelligent Systems (IS), 4-6 Sept. 2016, Sofia, Bulgaria., IEEE, 2016, ISBN: 978-1-5090-1354-8, DOI: 10.1109/IS.2016.7737486, 593-598, **2016**

Cited in::

3. Gocheva PV , Hinov NL, Gochev VP. "Generalized net based estimations on switching topologies in electronic circuits". Proceedings of the 44th International conference on applications of mathematics in engineering and economics: (AMEE’18). AIP Conference Proceedings, Volume: 2048, Issue: 1, pp. 060025-1–060025-6. American Institute of Physics, AIP Publishing, 2018 doi.org/10.1063/1.5082140, ISBN: 978-0-7354-1774-8, **2018**
4. Nedjalkov, I. "Characterization of the communication traffic generated by power electronic devices". Proc. of 2021 Int. Conf. Automatics and Informatics (ICAI), Varna, Bulgaria, pp. 362-365. ISBN: 978-1-6654-2662-6. IEEE, **2021**,

III. Tashev T., Monov V., Tasheva R.. High Performance Computations for Study the Stability of a Numerical Procedure for Crossbar Switch Node. In: Dimov I., Faragó I., Vulkov L. (eds) Numerical Analysis and Its Applications. NAA 2016., LNCS, volume 10187, Springer, Cham, 2017, ISSN: 03029743, DOI: 10.1007/978-3-319-57099-0_76, 665-673. SJR (Scopus):0.28, JCR-IF (Webof Science):0.402, **2016**

Цитира се в:

5. Hensel S., Marinov M. "Estimation of Magnetic Field Maps With Mobile Platforms". Proceeding of the 7th FDIBA Conference, Sofia, Bulgaria, 30-Nov. - 1 Dec.. 2017. TU-Sofia Publishinghouse, Sofia. Volume 1, pp.93-96. ISSN : 2535-132X, **2017**
6. Iliev, I.: Blagoev , I. "An Approach to Improve Web Video Streaming Security and Prevent Personal Data Leakage". Information&Security (ISU), vol.53, no.1, pp. 78-88. DOI 10.11610/isij.5306. ProconLtd., **2022**,

Bibliography

1. Bogdan V.V., Ivanov V.N. Informatization and technologization of social space [Conference]: Materials for the 1st International Symposium on Information Technologies. M. - Nizhny Novgorod, 1994, pp. 12-17. (in russian)
2. Colin K.K. Information problems of socio-economic development of society [Journal]: Problems of social informatics. Issue 1. Moscow: Soyuz Publishing House, 1995, p.5-9. (in russian)
3. Colin K.K. Information technology is a catalyst for the development of modern society [Journal]:. Journal "Information technologies", M., zero bray, 1995, p.2-8. (in russian)
4. David R. Theory of Discrete Control Devices [Book]: Moscow: Nauka, 1982. (in russian)
5. Buslenko N.P. Modeling of complex systems [Book]: M.: Nauka, 1975. 400 p. (in russian)
6. Ankudinov G.I. Synthesis of the structure of complex objects. Logic-combinatorial approach [Book]: L.: LGU, 1986. 258 p. (in russian)
7. Lesdon L. Optimization of large systems [Book]: M.: Nauka, 1975. 432 p. (in russian)
8. Voevodin V.V. Mathematical models and methods in parallel processes [Book]: M.: Nauka, 1986. 296 p. (in russian)
9. Collins G. Blay JM Structural Methods of System Development: From Strategic Planning to Testing [Book]: M.: Finance and statistics, 1986, 264 p. (in russian)
10. Solodovnikov VV, Tumarkin VI Theory of complexity and design of control systems [Book]: M.: Nauka, 1990. 166 p. (in russian)
11. Mesarovich M., Takahara J. General Systems Theory: Mathematical Foundations [Book]: M.: Mir, 1978. 311 p. (in russian)
12. Bozhko A.N. Methods of structural analysis in CAD [Journal]: Bulletin of MSTU. Ser. Instrumentation, 1991. N 2 p. 22-28 (in russian)
13. Odrin V.M., Kartavov S.S. Morphological analysis of systems [Book]: Kyiv: Naukova Dumka, 1977. 83 p. (in russian)
14. Kuhn S. VLSI Matrix Processors [Book]: M.: Mir, 1991. 672 p. (in russian)
15. Kotov V. E. Algorithms, software and architecture of multiprocessor computing systems [Book]: M.: Nauka, 1982. 336 p. (in russian)
16. Kotov V. E. Seti Petri [Book]: M.: Nauka, 1984. 160 p. (in russian)
17. Peterson J. Theory of Petri nets and system modeling [Book]: M.: Mir, 1984. 264 p. . (in russian)
18. Kron G. Study of complex systems in parts (diacoptics) [Book]: M.: Nauka, 1972, 544 p. (in russian)
19. Kron G. Tensor analysis of networks [Book]: M.: Sov. radio, 1978. 720 p. . (in russian)
20. Petrov A.E. Tensor methodology in systems theory [Book]:. Moscow: Radio and communication, 1985.152 p. (in russian)
21. Kulagin VP Tensor methods for designing structures of computing systems [Journal]: AVT. - 1989. N2. 64-71 p. (in russian)
22. Kulagin V.P. Algebra of network models for describing parallel computing systems [Journal]:. Automation and modern technologies. 1993. N2, 25 - 30 p. (in russian)
23. Atanassov K. Generalized Nets [Book]: World Scientific, Sing., N.J., London, 1991.
24. Atanassov K. Generalized Net sand System Theory [Book]: Akad. Press “Prof.M.Drinov”, Sofia, Bulgaria, 1997.
25. Atanassov K. Generalized nets in Articial Inteligence [Book]: Vol.1: Generalized Netsand Expert Systems, Academic Publishing House “Prof. M. Drinov”, Sofia, 1998.
26. Atanassov K., H. Aladjov, Generalized nets in Articial Inteligence [Book]: Vol.2: Generalized nets and Macchine Learning, Academic Publ. House “Prof. M. Drinov”, Sofia, 2000.
27. Atanassov, K., On Generalized Nets Theory [Book]:. B.A.Monographs (11) Prof. Marin Drinov Publishing House ofthe B.A.S., Sofia, 2007.
28. Atanasov K., E. Sotirova. Aggregated Networks [Book]: Academic publishing house "|Prof. M. Drinov", Sofia, 2017

29. Shahpazov G., Doukovska L. Generalized net model of internal financial structural unit's functionality with intuitionistic fuzzy estimations [Conference]: Proc. of the 17th International Conference on Intuitionistic Fuzzy Sets, 2013, Sofia, Bulgaria. Notes on Intuitionistic Fuzzy Sets (NIFS), vol. 19, №3, pp. 111-117, 2013.
30. Zoteva, D., Angelova, N. (2021). Generalized Nets. An Overview of the Main Results and Applications [Journal]: In: Atanassov, K.T. (eds) Research in Computer Science in the Bulgarian Academy of Sciences. Studies in Computational Intelligence, vol 934 pp.177-226. Springer, Cham. https://doi.org/10.1007/978-3-030-72284-5_10 . 2021
31. Stratiev D.D., Stratiev D., Atanassov, K. Modelling the Process of Production of Diesel Fuels by the Use of Generalized Nets [Journal]: MATHEMATICS, Volume 9, Issue 19, Article Number 2351. MDPI, Basel, Switzerland. 2021
32. Atanassov K.T., Vassilev P., et all. Generalized Net Model of Forest Zone Monitoring by UAVs. [Journal]: MATHEMATICS, Volume 9, Issue 22, Article Number 2874. MDPI, Basel, Switzerland. 2021
33. Lubich M., et all. A Generalized Net Model of the Prostate Gland's Functioning [Journal]: MATHEMATICS, Volume 10, Issue 3, Article Number 479. MDPI, Basel, Switzerland. 2022
34. Chao J., B. Liu. High performance switches and routers [Book]: John Wiley & Sons, 2007
35. Mirchev S. Switching in communication networks [Book]: New knowledge, Sofia, 2010
36. Elhanany I., M. Hamdy. High-performance Packet Switching Architectures [Book]: Springer-Verlag, 2007
37. Deb S., Shah D., Shakkottai S. Fast Matching Algorithms for Repetitive Optimization: An Application to Switch Scheduling [Conference]: Information Sciences and Systems, 2006 40th Ann. Conference on, 22-24 March 2006, pp. 1266-1271. 2006
38. Chen W., J. Mavor, P. Denyerand D. Renshaw, Traffic routing algorithm for serial superchip system customization [Journal]: IEE Proc.137:[E]1, 1990.
39. Meng, J., Gebara N., Ho-Cheung Ng, Costa P., Luk W. Investigating the Feasibility of FPGA-based Network Switches [Conference]:. Application-specific Systems Architectures and Processors (ASAP) 2019 IEEE 30th Int. Conf. on, vol. 2160-052X, IEEE Publ., pp.218-226, 2019
40. Rojas-Cessa R. Interconnections for Computer Communications and Packet Networks [Book]: CRC Press, 2017.
41. Alcoz A. G., A. Dietmüller and L. Vanbever, SP-PIFO: Approximating Push-In First-Out Behaviors using Strict-Priority Queues [Conference]: Proceedings of USENIX Symposium on Networked Systems Design and Implementation*,* pp. 59-76, 2020.
42. Li X., *et al.,* RPQ: Resilient-Priority Queue Scheduling for Delay-Sensitive Applications [Conference]: 2022 IEEE 23rd International Conference on High Performance Switching and Routing (HPSR)*,* Taicang, Jiangsu, China, 2022, pp. 53-58, 2022
43. Li T., Y. Cai, Joint Routing and Scheduling for Deterministic Networking: A Segment Routing Approach [Conference]:, 2022 IEEE 23rd International Conference on High Performance Switching and Routing (HPSR)*,* Taicang, Jiangsu, China, 2022, pp. 189-194, 2022
44. Guo Y., J. Chen, K. Huang and J. Wu, A Deep Reinforcement Learning Approach for Deploying SDN Switches in ISP Networks from the Perspective of Traffic Engineering [Conference]:, 2022 IEEE 23rd International Conference on High Performance Switching and Routing (HPSR)*,* Taicang, Jiangsu, China, 2022, pp. 195-200, 2022
45. Tao J., S. Liu and C. Liu, A Traffic Scheduling Scheme for Load Balancing in SDN-Based Space-Air-Ground Integrated Networks [Conference]:, 2022 IEEE 23rd International Conference on High Performance Switching and Routing (HPSR)*,* Taicang, Jiangsu, China, 2022,
46. Vabishchevich P. VFort. http://www.nomoz.org/site/ 629615/vfort.html (last checked September 16, 2022). http://www.imamod.ru/~vab/vfort/download.html . 2022