INSTITUTE OF INFORMATION
AND COMMUNICATION
TECHNOLOGIES
BULGARIAN ACADEMY OF
SCIENCE

1869

A Numerical Study of the Upper Bound of the Throughput of a Crossbar Switch Utilizing MiMa-algorithm

Tasho Tashev, Vladimir Monov

{ttashev,vmonov}@iit.bas.bg

NMA 2014, 20-24 August 2014, Borovets, Bulgaria

Abstract

The efficiency of the switch performance is firstly evaluated by the throughput (THR) provided by the switch node.

In the present paper we propose an extension of the family of patterns for hotspot load traffic simulating. The results from the computer simulations of the throughput (THR) of a crossbar packet switch with these patterns are presented. The necessary computations have been performed on the grid-cluster of IICT-BAS.

Our simulations utilize the MiMa-algorithm for non-conflict schedule, specified by the apparatus of Generalized Nets. A numerical procedure for computation of the upper bound of the throughput is utilized.

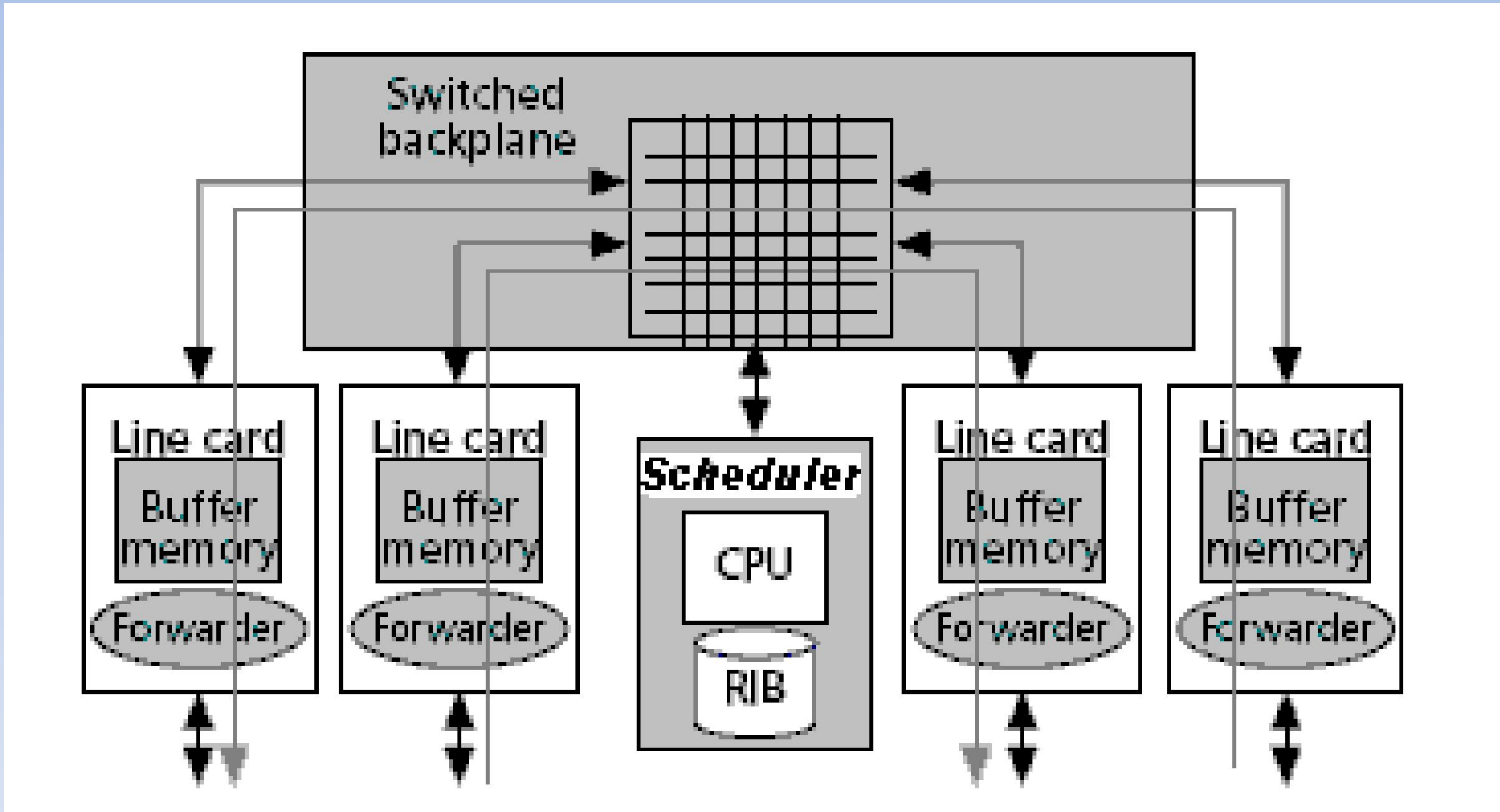
It is shown that the throughput of the MiMa-algorithm with the suggested family of patterns tend to 100 %.

Outline

1. Introduction
2. Algorithm and simulations
3. Numerical procedure for computation of upper bound
4. Upper bound for MiMa-algorithm
5. Conclusion

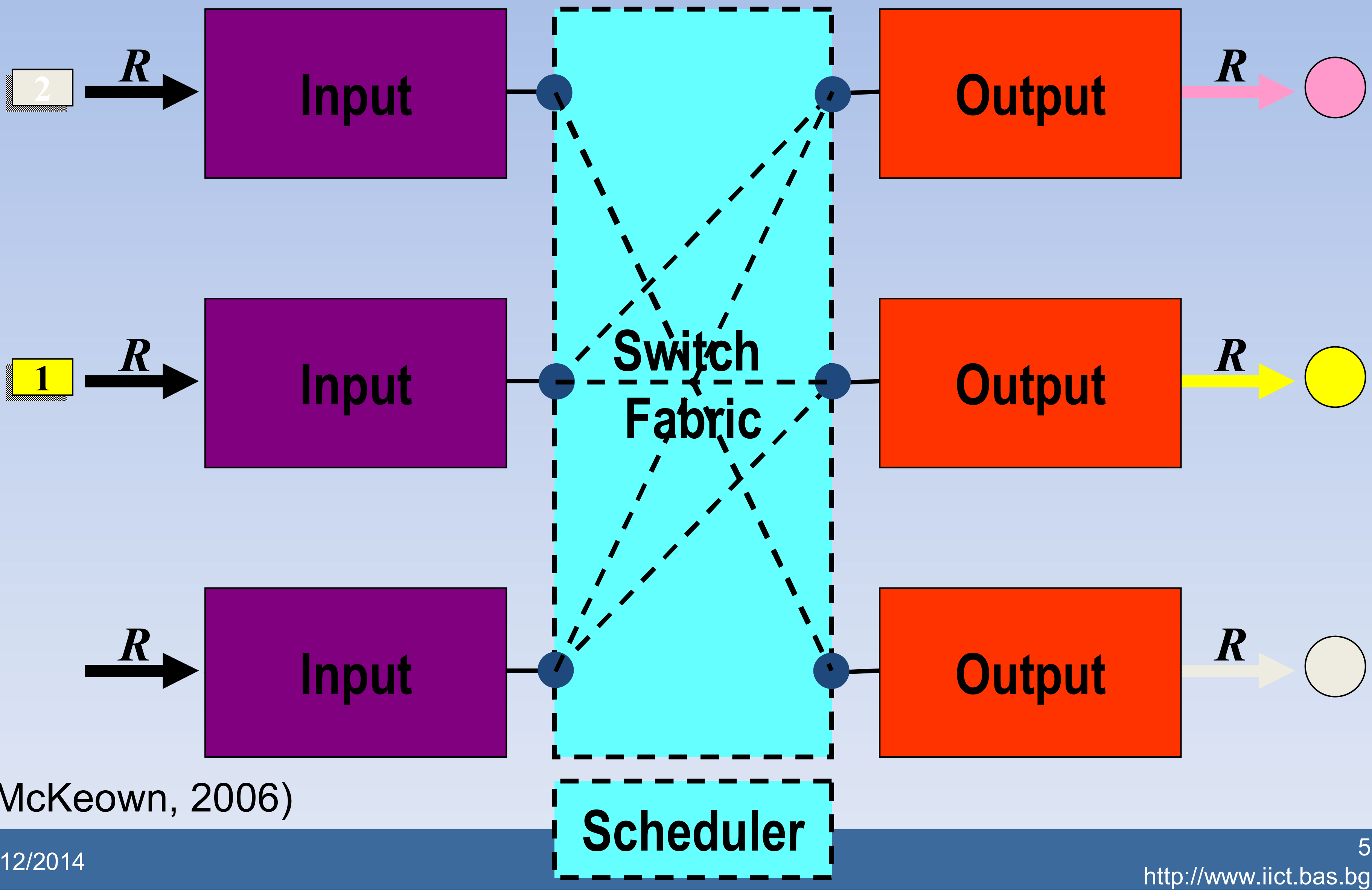
1. Introduction

Crossbar switch node 3-th generation:



Csaszar, A. et all, IEEE Network, 2007, Vol. 4

1. due to randomly incoming traffic the output must be controlled



(McKeown, 2006)

1. non-conflict schedule

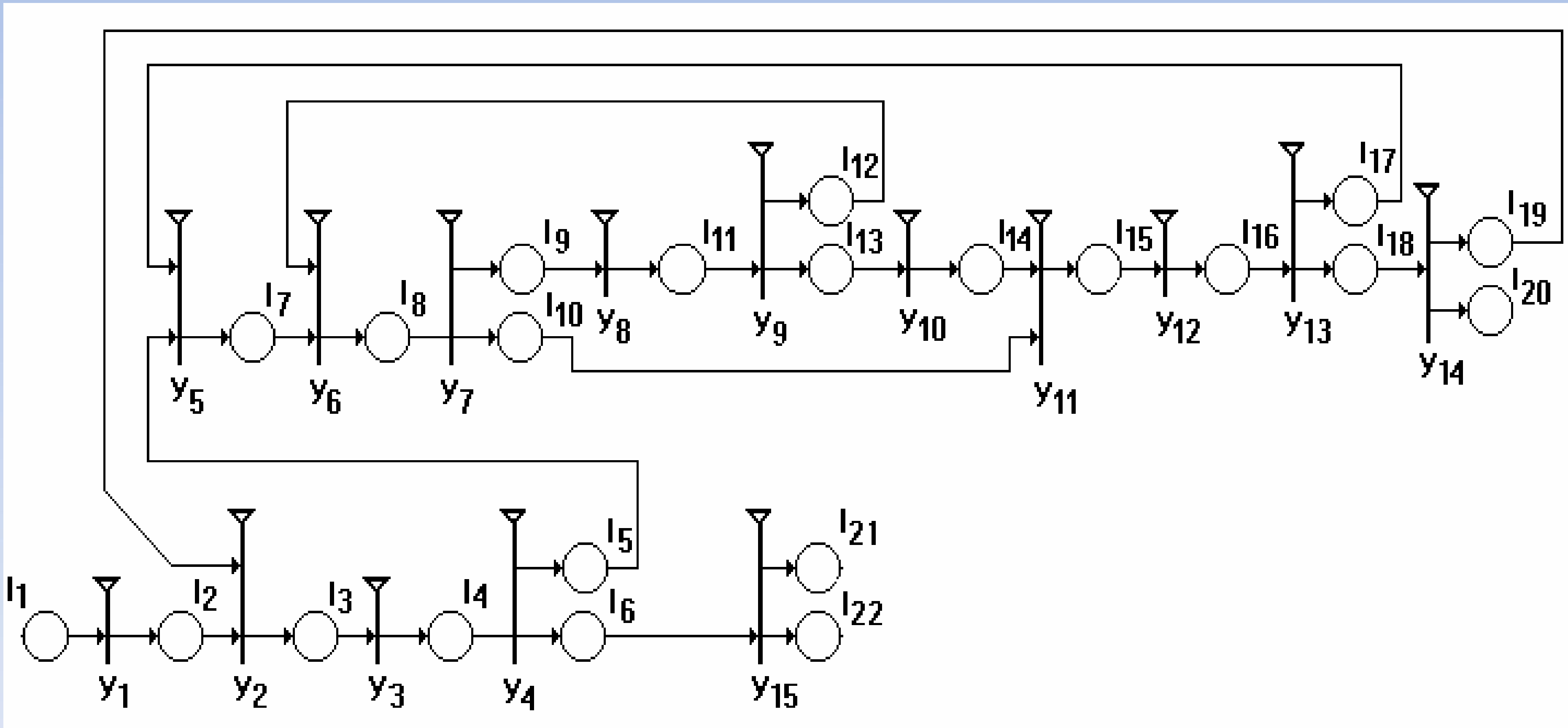
- Due to randomly incoming traffic switching of commutation field must be controlled and scheduled to eliminate conflicts at the crossbar fabric.
- The goal of the traffic-scheduling for the crossbar switches is to
  - maximize the throughput of packet through a switch node;
  - minimize packet blocking probability;
  - minimize packet waiting time.

An attempt to reach simultaneously these three goals leads to problems with non-polinomial completeness of the solution. (NP-complete).

Some solutions which satisfy the goals partially are suggested (algorithms for schedule calculation) : PIM, iSLIP, BvN, DISQUO, StablePlus, CTC(N)…

2. Algorithm and simulations

Generalized nets form of MiMa-algorithm for non-conflict schedule suggested by us (Tashev-2009) is shown :

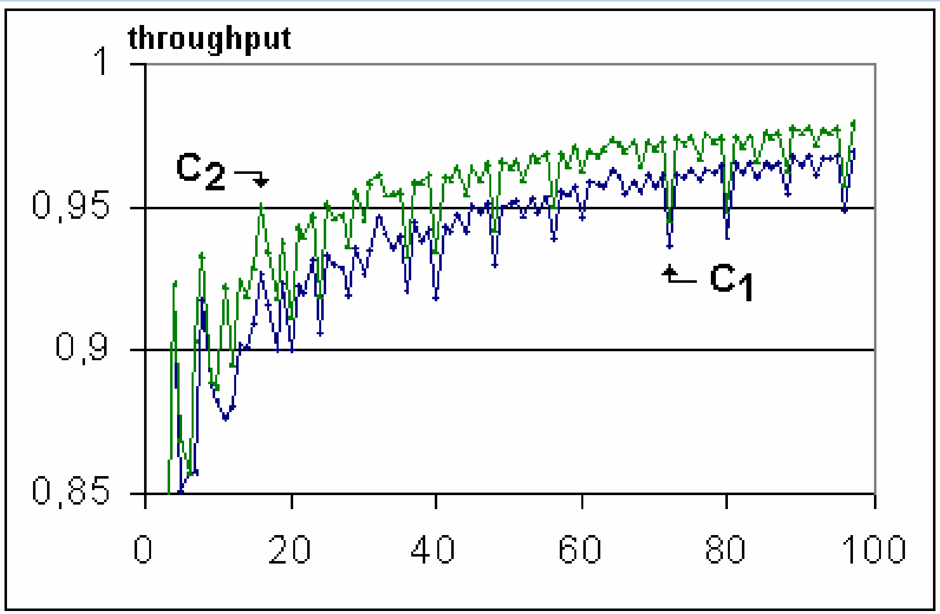


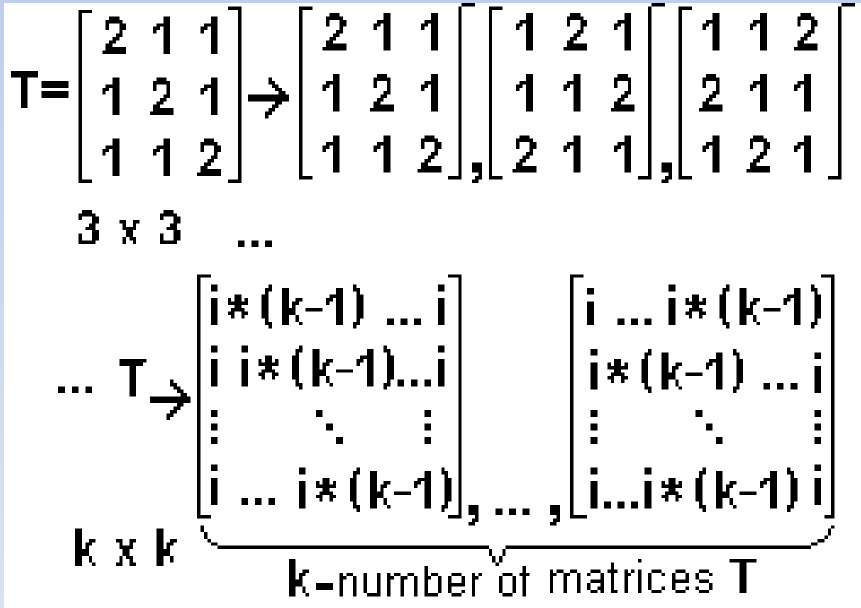
Initial characteristics for token in l_1 : $ch_0 = (pr_1ch_0, pr_2ch_0) = (n, T)$.

2. conditions for the simulation

We performed simulations for a specific algorithm for non-conflict schedule, a model for incoming traffic and load intensity. Our modeling of the throughput utilizes MiMa-algorithm [8], Chao-model for hotspot load traffic matrix *T* [9] and ρ=100% load intensity of each input (*i.i.d.* Bernoulli) [10]. In this case the throughput of crossbar node increases (to a certain limit?). In case of size *n* of the switch field: $n \in [3,97]$ and scale *i* of input buffers loading for expanded family of patterns for Chao-model: Chao-*i*, $i \in [1,2]$ the results are shown in figure to the left.

$$THR = r_{opt}/r_{sim},\ r^{Chao}_{opt} = i.2.(k-1)$$





THR for MiMa-algorithm with Chao-1,Chao-2. Traffic matrices for Chao-*1* and Chao-*i*.

3. Numerical procedure

We perform simulations for a specific algorithm for non-conflict schedule, a model for incoming traffic and load intensity.

We choose the interval for values of n and i; where i will define the increase in size of the input buffer. As a result, we have a set of curves for selected values of $n \in [n1, n2]$; and $i \in [i1, i2]$. $n1 \geq 2$, $i1 \geq 1$. Let us chose values for i :

$$i = 1, m_1, m_2, m_3, \ \dots, m_p, \quad \text{where } 1 = m_0 < m_1 < m_2 < m_3 < \dots < m_p$$

We perform $(p + 1)$ simulations in order to obtain $(p + 1)$ curves for THR. The obtained curves will be denoted as follows:

$$f_1(n, i) = f(n,m_0); \qquad f_2(n, i) = f(n,m_1), \qquad \dots \qquad , f_{p+1}(n, i) = f(n,m_p)$$

Denote the difference between two consecutive curves fj and $fj+1$ by $resj$:

$$res_1(n, i) = f_2(n, i) - f_1(n, i) = f(n,m_1) - f(n,m_0)$$

$$\vdots$$

$$res_p(n, i) = f_{p+1}(n, i) - f_p(n, i) = f(n,m_p) - f(n,m_{p-1})$$

Denote the ratio of the values of two successive curves res_j and res_{j+1} through $\boldsymbol{\delta}_j$:

$$\boldsymbol{\delta}_1(n, i) = res_2(n, i) / res_1(n, i) = (f(n,m_2) - f(n,m_1)) / (f(n,m_1) - f(n,m_0))$$

$$\vdots$$

$$\boldsymbol{\delta}_{p-1}(n, i) = res_p(n, i) / res_{p-1}(n, i) = (f(n,m_p) - f(n,m_{p-1})) / (f(n,m_{p-1}) - f(n,m_{p-2}))$$

3. Computation of ratio δ

Simulation data allow us to calculate $\boldsymbol{\delta}_1, \boldsymbol{\delta}_2, \ldots, \boldsymbol{\delta}_{p-1}$.

If we can find a dependency $\boldsymbol{\delta}_{j+1} = \varphi(\boldsymbol{\delta}_j)$ for $\boldsymbol{\delta}_1, \boldsymbol{\delta}_2, \ldots, \boldsymbol{\delta}_{p-1}$ in the case $j \to \infty$, then we can determine the expected upper bound.

From the last formula we obtain:

$$f_{p+1}(n, i) = f(n,m_{p-1}) + \boldsymbol{\delta}_{p-1}(n, i).(f(n,m_{p-1}) - f(n,m_{p-2}))$$

and for a known dependency $\boldsymbol{\delta}_{j+1} = \varphi(\boldsymbol{\delta}_j)$, we can write

$$f_{p+2}(n, i) = f(n,m_{p-1}) + [1 + \varphi(\boldsymbol{\delta}_{p-1}(n, i))] \,.\, \boldsymbol{\delta}_{p-1}(n, i).(f(n,m_{p-1}) - f(n,m_{p-2}))$$

$$\vdots \quad \vdots \quad \vdots$$

$$f_{p+q}(n, i) = f(n,m_{p-1}) + [1 + \varphi(\boldsymbol{\delta}_{p-1}(n, i)) + \varphi(\boldsymbol{\delta}_{p-1}(n, i)),\varphi(\boldsymbol{\delta}_{p}(n, i)) + \ldots$$
$$\ldots+\varphi(\boldsymbol{\delta}_{p-1}(n, i)),\varphi(\boldsymbol{\delta}_{p}(n, i)) \ldots \varphi(\boldsymbol{\delta}_{p+q-3}(n, i))] \,.\boldsymbol{\delta}_{p-1}(n, i).(f(n,m_{p-1})-f(n,m_{p-2}))$$

When $q \to \infty$ then $f_{p+q \to \infty}(n, i)$ is the necessary bound $\lim_{i \to \infty, n = const} f(n, i)$,

If there is an upper bound of the throughput of a switch node, it is clear that the dependency $\boldsymbol{\delta}_{j+1} = \varphi(\boldsymbol{\delta}_j)$ exists. Then the sum:

$$[1 + \varphi(\boldsymbol{\delta}_{p-1}(n, i)) + \varphi(\boldsymbol{\delta}_{p-1}(n, i)),\varphi(\boldsymbol{\delta}_{p}(n, i)) + \ldots$$
$$\ldots+\varphi(\boldsymbol{\delta}_{p-1}(n, i)),\varphi(\boldsymbol{\delta}_{p}(n, i)) \ldots \varphi(\boldsymbol{\delta}_{p+q-3}(n, i))]$$

for $q \to \infty$ it is convergent and has a boundary (finite).

3. Finding dependencies $\delta_{j+1}=\varphi(\delta_j)$

We have found one such relation for our model of PIM-algorithm, specified by means of Generalized nets [6], with Chao-model for hotspot load traffic, for which we defined the family of patterns Chao_i for traffic matrices [7]. For a simulation with this family of patterns we have chosen the sequences for *i* : $i = 1, m^1, m^2, m^3, \ldots, m^p, \ldots,$

The minimal value of m in its definition area $m \in [2,3,4, \ldots,)$ is m=2. In the interval $m \in [2,3,4,5]$ we found that the dependence $\boldsymbol{\delta_{j+1}=\varphi(\delta_j)}$ is a constant, i.e.

$$\boldsymbol{\delta_{j+1}} = \boldsymbol{\delta_j} = m^{-1/2}$$

with an accuracy depending on the error of simulations.

As a consequence, the upper boundary in case $m = const$ can be calculated as:

$$f_{p+1}(\mathbf{n},i) = f(\mathbf{n},m^{p-1}) + \delta(m) \,.\, (f(\mathbf{n},m^{p-1}) - f(\mathbf{n},m^{p-2}))$$

$$f_{p+2}(\mathbf{n},i) = f(\mathbf{n},m^{p-1}) + (\delta(m)+ \delta^2(m)).\, (f(\mathbf{n},m^{p-1}) - f(\mathbf{n},m^{p-2}))$$

$$\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots$$

$$f_{p\to\infty}(\mathbf{n},i) = f(\mathbf{n},m^{p-1}) + (\delta(m)+\delta^2(m)+\ldots+ \delta^p+\ldots) \,.\, (f(\mathbf{n},m^{p-1}) - f(\mathbf{n},m^{p-2}))=$$
$$= f(\mathbf{n},m^{p-1}) + (m^{-1/2}+(m^{-1/2})^2+\ldots+(m^{-1/2})^p +\ldots) \,.\, (f(\mathbf{n},m^{p-1}) - f(\mathbf{n},m^{p-2})) =$$
$$= f(\mathbf{n},m^{p-1}) + (m^{1/2}-1)^{-1} \,.\, (f(\mathbf{n},m^{p-1}) - f(\mathbf{n},m^{p-2}))$$

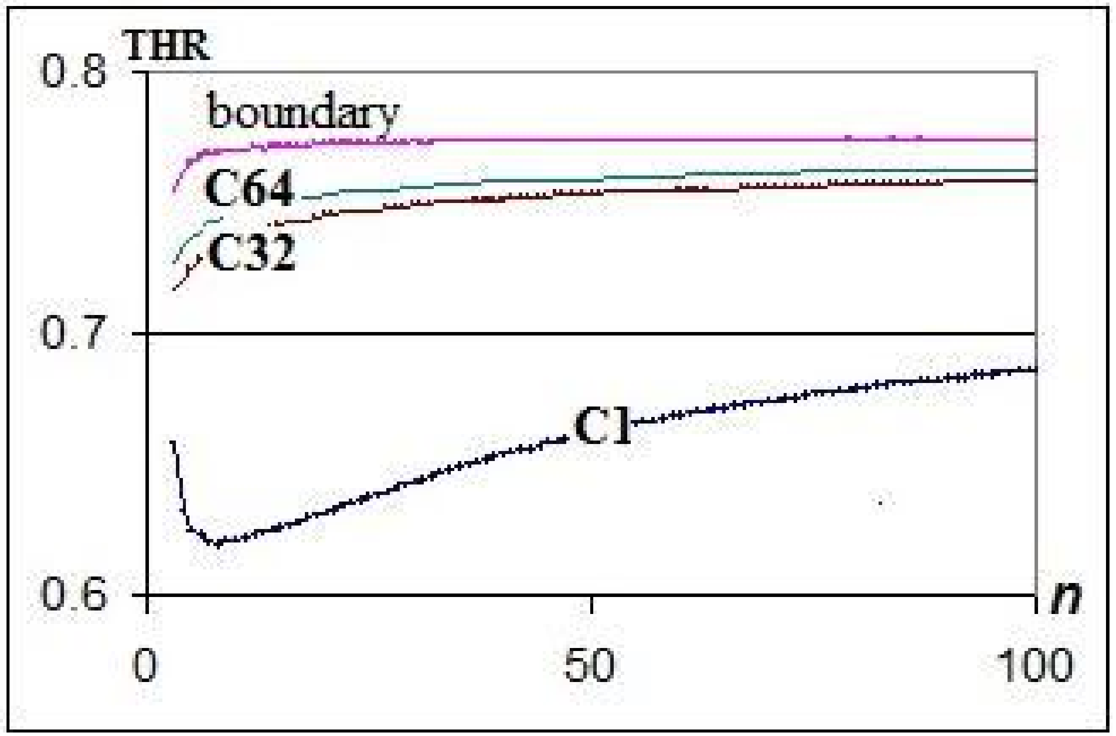
We know from the theory that an infinite number series of the form $1/a + 1/a^2 + 1/a^3 + \ldots + 1/a^i + \ldots$, where $a > 1$, converges as $i \to \infty$ to the value of $1/(a-1)$. In case m=2 we assume that the parameter of convergence **δ** forms a series with a = 1.4142.

3. calculating the boundary

In this simulation (m=2) we calculate the boundary of PIM-algorithm by the formula:

$$f_{p \to \infty}(n,i) = f(n, 64) + [(2^{1/2} - 1)^{-1}].(f(n, 64) - f(n, 32))$$

The result is shown in Figure.



The differences between the obtained during the simulations values of δ_i and the value $m^{-½}$ are equal to the absolute error δ. If we accept $\delta = m^{-½}$, the error of $f_{p \to \infty}$ will decrease at least twice.

Thus we conclude that $\lim_{i \to \infty, n \to \infty} f(n, i) = 0{,}775 \pm 0{,}001$ for PIM-algorithm.

4. Upper boundary for MiMa

If the dependence $\delta_{j+1} = \delta_j = m^{-1/2}$ in the above described procedure is not a particular one (only for PIM-algorithm), we can use the procedure for calculation of upper bound of THR for MiMa-algorithm.

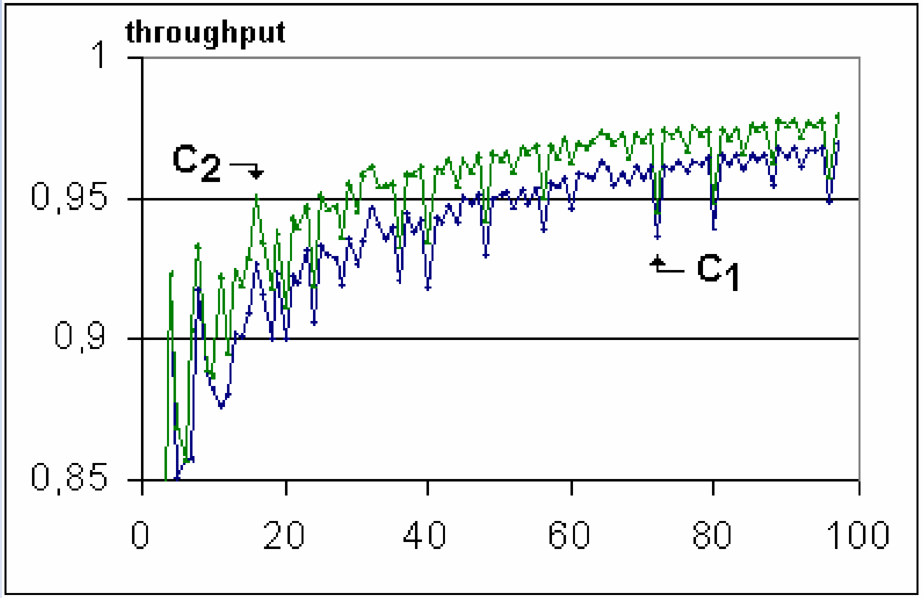
For this goal we have chosen the sequences for *i* : $i = 1, m^1, m^2, m^3, \dots, m^p, \dots$

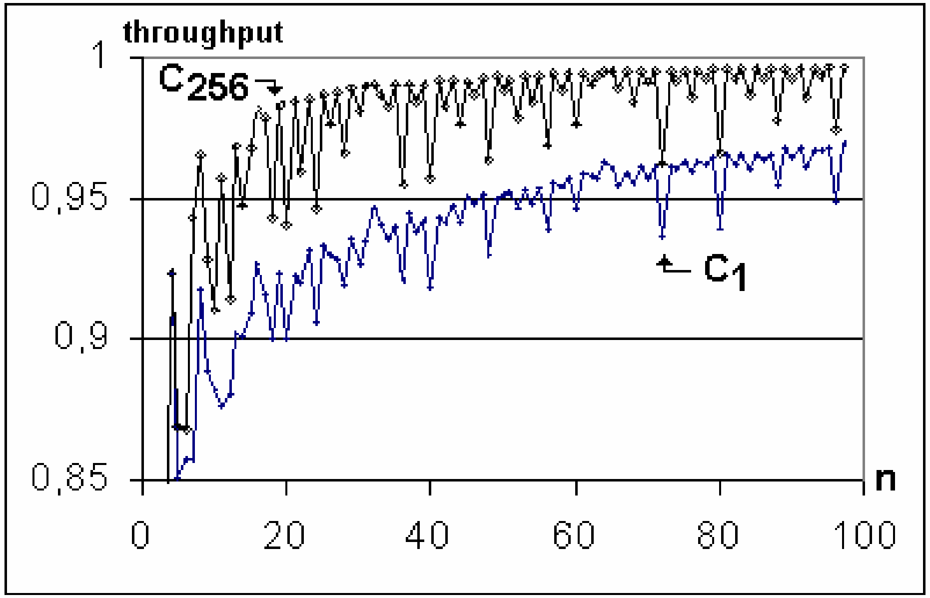
for simulations of THR for MiMa-algorithm, with value m=2.

The initial evaluation of the required number of curves for THR is at least 4 (from Pattern *Chao*1).

We get results for Chao_i : *C*1,*C*2,C4, *C*8,*C*16,*C*32,*C*64,C128,C256 which are shown in Figure left.

The dimension *n* varies from 3x3 to 97x97 and n simulations for each pattern.

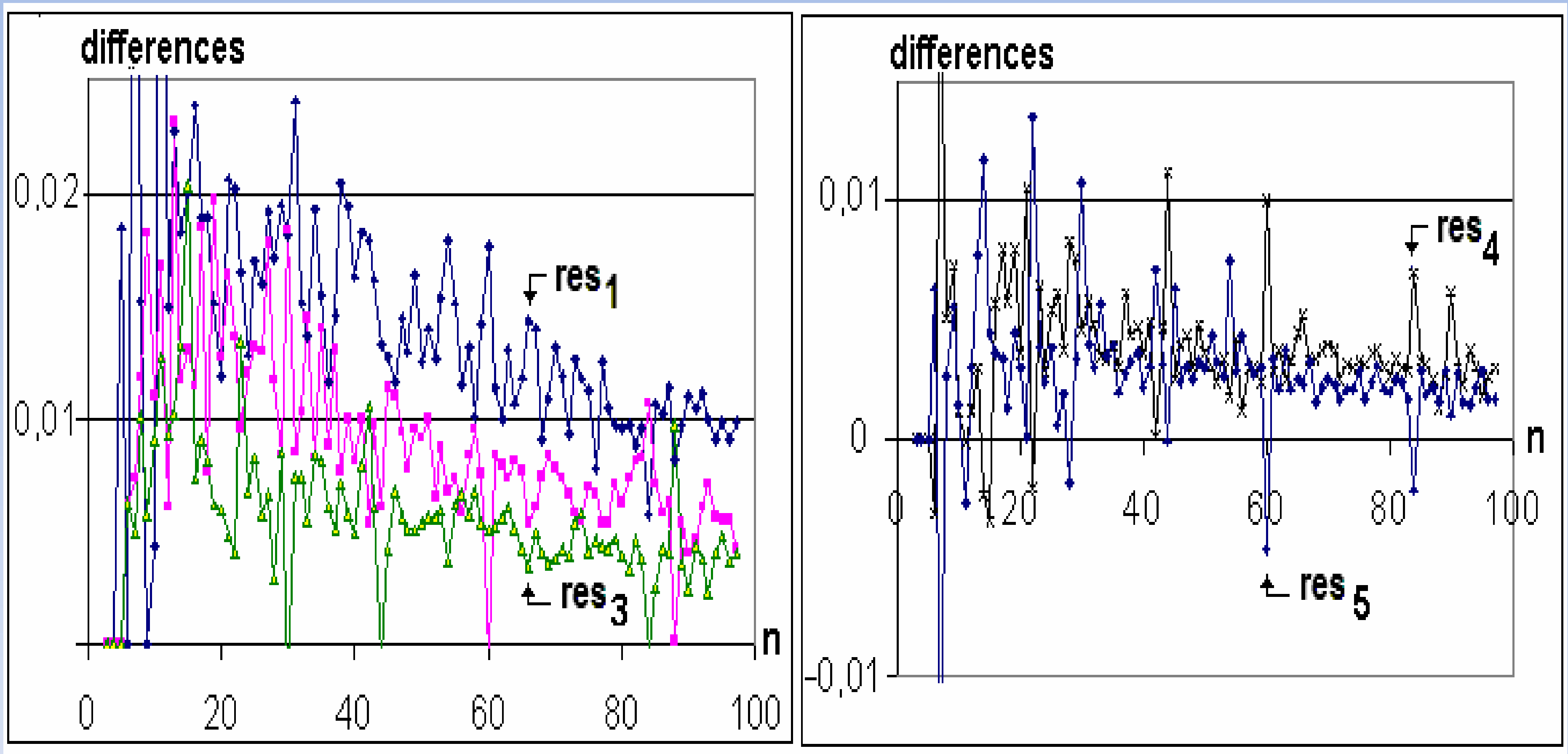




Throughput for MiMa-algorithm with Chao-traffic.

4. calculation of differences

Then we calculate the difference between throughput for neighboring patterns.

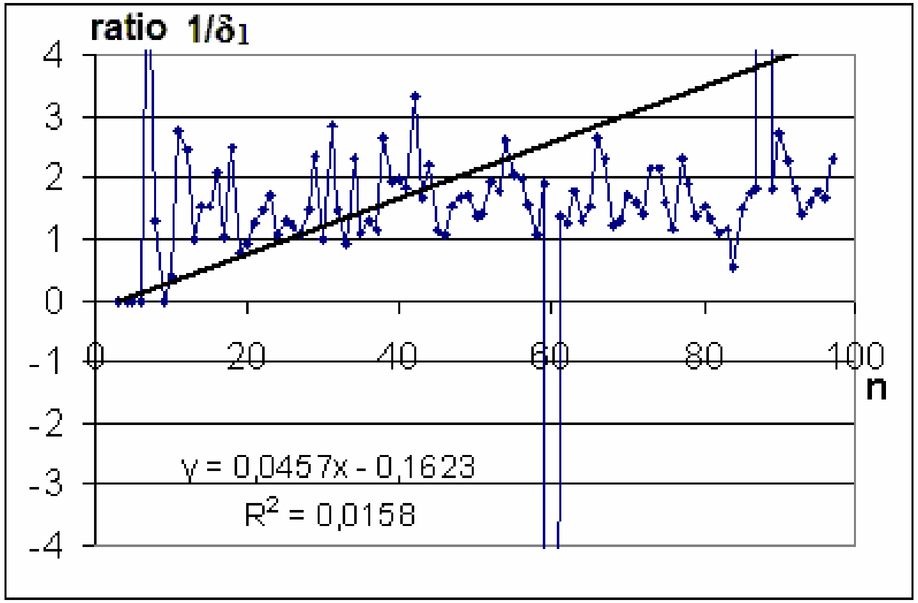


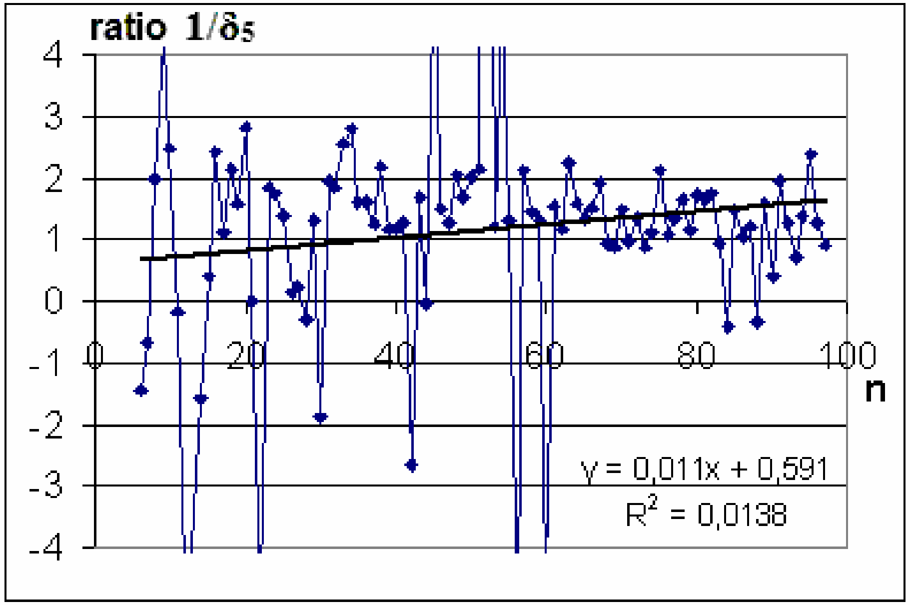
The obtained curves for the differences res1, … ,res5.

Then we calculate the convergence parameter δ_j which is the ratio of the differences.

4. calculating of ratio δ

We can accept that the values of $\boldsymbol{\delta}_j$ oscillate around $2^{-1/2}$.





Ratio ($1/\delta_1$) between differences

Ratio $1/\delta_5$ between differences

In this simulation (m=2) we calculate the boundary by the formula:

$$f_{p\to\infty}(n,i) = f(n, 64) + [(2^{1/2} - 1)^{-1}].(f(n, 64) - f(n, 32))$$

This choise is for $\boldsymbol{\delta}_5$ - it has the least deviation from $m^{-1/2}$.

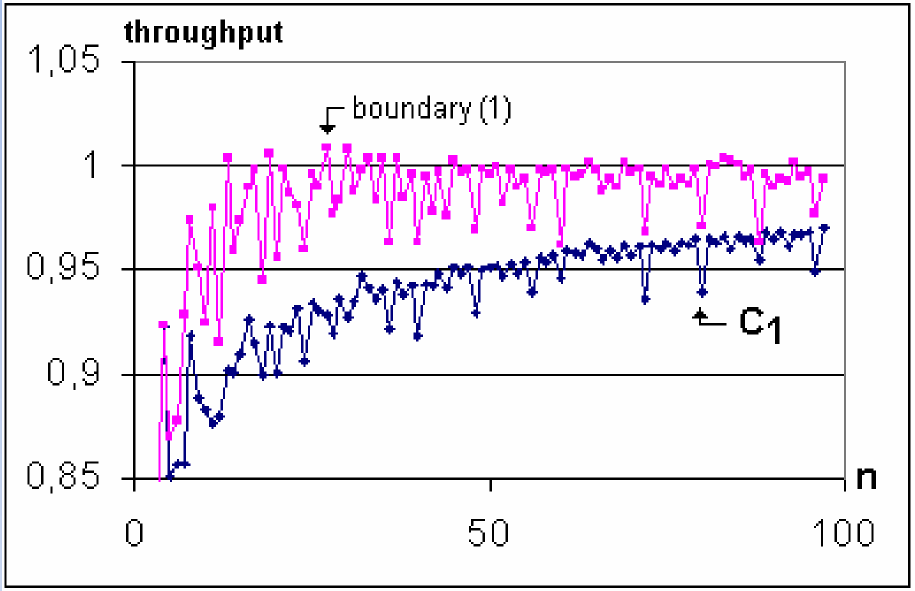
For comparison is shown a boundary which is calculated about $\boldsymbol{\delta}_1$:

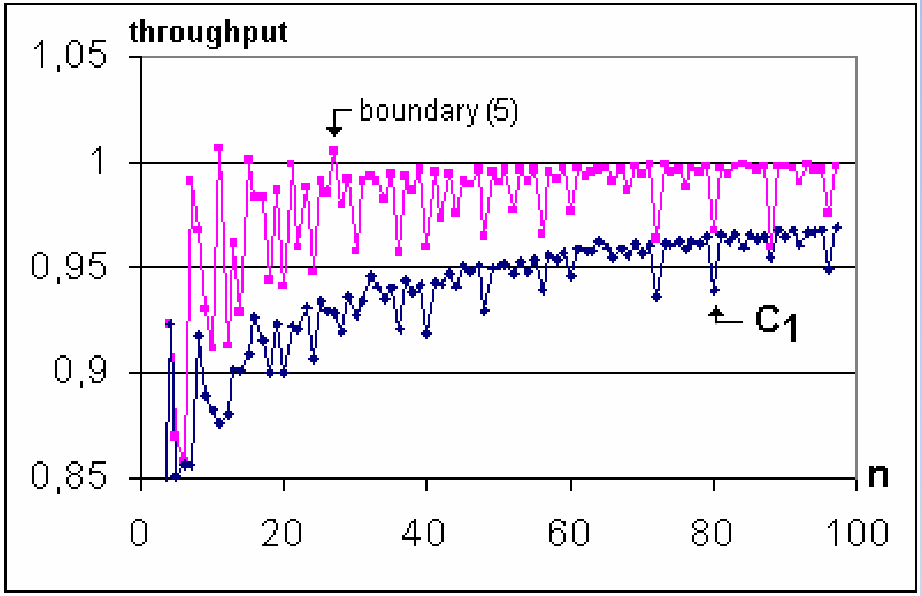
$$f_{p\to\infty}(n,i) = f(n, 4) + [(21/2 - 1) -1].(f(n, 4) - f(n, 2))$$

Thus we conclude that $\lim_{i\to\infty,\, n\to\infty} f(n, i) = 1$ for MiMa-algorithm.

4. calculating the boundary

The result is shown in Figure.





Upper boundary of throughput for δ_1 Upper boundary of throughput for δ_5

The differences between the values of δ_i obtained in the simulation and the value $\delta(m)= m^{-½}$ are a measure for simulation accuracy.

Therefore for calculation of the upper bound we chose these two successive curves f_j and f_{j+1} for which δ_j has the least derivation from *m-½* .

5. Conclusion

Our computer simulation confirms applicability of the suggested procedure with extended family of patterns for hotspot load traffic.

The obtained results give an upper bound of the THR for $i \in [3, 97]$ which enables us to estimate the limit of the THR of MiMa-algorithm for $n \to \infty$. This estimate is obtained to be 100%.

In a future study, the suggested modification will be tested using m=3,4,5 for hotspot and with other models of the incoming traffic, for example unbalanced traffic models.

Acknowledgments

The research work reported in the paper is partly supported by the project AComIn "Advanced Computing for Innovation", grant 316087, funded by theFP7 Capacity Programme (Research Potential of Convergence Regions).

Thank you!