Performance Evaluation of SMM Clos-Network Switches under Static Connection Patterns Scheme

Janusz Kleban
Faculty of Electronics and Telecommunications
Poznan University of Technology
Poznań, Poland
janusz.kleban@put.poznan.pl

Jarosław Warczyński
Faculty of Electrical Engineering
Poznan University of Technology
Poznań, Poland
jaroslaw.warczynski@put.poznan.pl

***Abstract*—This paper is devoted to performance evaluation of the SMM (*Space-Memory-Memory*) Clos-network switch under a packet dispatching scheme employing static connection patterns, called SD (*Static Dispatching*). The control algorithm with static connection patterns can be easily implemented in the SMM fabric due to bufferless switches in the first stage. One of the very important performance factor of packet switching nodes is the stability. In general, a switch is stable for a particular arrival process if the expected length of the packet queues does not grow without limits. In this paper we use the second Lapunov method to prove the stability of SMM Clos-network switches under SD packet dispatching scheme. Results of simulation experiments, in terms of average cell delay and packet queue lengths are also shown.**

Keywords—Clos-network switch, packet dispatching algorithms, packet switching network, stability of switching network

I. INTRODUCTION

Connecting paths between input and output ports in switches/routers are provided by switching fabrics, which are the main part of every packet switching nodes. The switching fabrics replace too slow buses mainly in middle-size and high-end routers and switches. They can establish connections between input ports and requested output ports, and simultaneously transmit packets. The one stage switching fabrics called crossbar switches are used mainly in middle-size routers/switches [1]. Basically, an $N \times N$ crossbar switch consists of a square array of N^2 individually operated crosspoints *(N* represents the number of inputs and outputs). Each crosspoint has two possible states: cross (default) and bar, and corresponds to input-output pair. A connection between input port i and output port j is established by setting the (i, j)th crosspoint to the bar state while letting other crosspoints along the connecting paths remain in the cross state. The crossbar switch can transfer up to N cells from different input ports to different output destinations in the same time slot. The control algorithm for the crossbar fabric is very simple due to the bar state of the crosspoint can be triggered individually by each incoming packet when its destination matches with the output address. The crossbar fabrics are complex in terms of the crossponts number, which grows as N^2. The arbitration process that has to choose packets to be sent from inputs to outputs in each time slot can also become a system bottleneck as the switch size increases.

In high-end routers multi-stage or even multi-stage and multi-plane switching fabrics are used. In this case the Clos-network switches are very attractive because of its modular and scalable architecture. The Clos-network fabric is composed of crossbar switches arranged in stages [2]. This switching fabric is currently used by network equipment vendors to build core routers e.g. Cisco's CRS series, Juniper's T series, and Brocade's BigIron RX Series. For example, in the CISCO's new router called CRS-X (*Carrier Routing System - X*), a multi-stage and multi-plane switching fabric is used. This family of routers focus on extreme scale. One standard 7 ft rack chassis of CRS-X deployment can deliver up to 12.8 terabits per second. The system can be clustered together in a massive configuration of up to 72 chassis, which would deliver up to 922 Tbps of throughput [3].

High-speed switching fabrics adopt the use of cells, fixed-length data units. All incoming variable-length packets (e.g. IP packets) are segmented at ingress line cards into fixed-size cells. Next, they are transmitted in time slots through the switching fabric, and re-assembled into packets at egress line cards, before they depart [1]. While a cell is being routed in a packet switching system, it can face a contention problem resulting from the fact that two or more cells compete for a single resource. Cells that have lost contention must be either discarded or buffered. According to buffer allocation schemes Clos-network packet switches are classified to: Space-Space-Space (SSS or S^3), Memory-Memory-Memory (MMM), Memory-Space-Memory (MSM), and Space-Memory-Memory (SMM) switches.

In this paper, we analyze the SMM Clos-network switch [4], where bufferless modules are used in the first stage and buffered crossbars in the second and the third stages. Due to bufferless modules in the first stage very simple control algorithm may be implemented to distribute cells to the central modules e.g. static dispatching (SD).

The remainder of this paper is organized as follows. Section II introduces some background knowledge concerning the SMM Clos-network switch and the SD algorithm. Using Lyapunov second method, we prove that the investigated

switching fabric is stable under the SD packet dispatching scheme in Section III. Section IV presents simulation results obtained for the SD scheme. We conclude this paper in section V.

II. The SMM Clos-Switching Fabric and SD Scheme

The three-stage Clos switching fabric architecture is denoted by *C(m, n, r)*, where the parameters *m, n*, and *r* entirely determine the structure of the network. There are *r* input modules (IM) of capacity $n \times m$ in the first stage, *m* central modules (CM) of capacity $r \times r$, and *r* output modules (OM) of capacity $m \times n$ in the third stage. The capacity of this switching system is $N \times N$, where $N = nr$. The three-stage Clos-network switch is strictly non-blocking if $m \geq 2n-1$ and rearrangeable non-blocking if $m \geq n$.

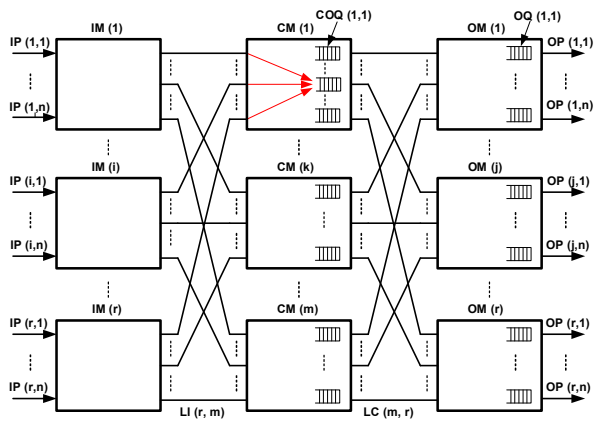

Fig. 1. The SMM Clos-network switch

In the basic SMM Clos-network switch (shown in Fig. 1), the first stage consists of *r* bufferless IMs with *n* input ports (IPs) each. The second stage consists of *m* CMs, and each of them has *r* FIFO buffers (COQs), one per output. Maximum *r* cells from *r* IMs may arrive to one COQ buffer, so it must work *r* times faster than the line rate. The third stage consists of *r* OMs, where each output port *OP(j, h)* has FIFO output buffer (OQ). Maximum *m* cells from *m* CMs may arrive to one OQ, so to store all cells during one time slot it must work *m* times faster than the line rate. The interstage links between IMs and CMs are denoted by $L_I(i, k)$, where *i* represents the number of IM, and *k* - the number of CMs, whereas $L_C(k, j)$ denotes interstage links between CM(*k*), and OM(*j*). Instead of using shared-memory CM and OM modules it is possible to employ the CQ (*Crosspoint Queued*) switches, where the speed-up is not necessary [5].

The SD scheme investigated in this paper seems to be the simplest packet dispatching algorithm that can be implemented in the SMM Clos-network switch. It is adaptation of the SRRD (*Static Round-Robin Dispatching*) to the SMM Clos-network switch, and is less demanding in terms of hardware, in comparison with other proposed schemes (e.g. [6]). The SD scheme does not need any special arbitration e.g. the handshaking processes, to distribute cells to the CMs. The key idea of the scheme are static connection patterns which are used in each IM. The consecutive static connection patterns used in IMs are shown in Fig. 2.

The connection patterns are the same in all IMs and are shifted to the next one in consecutive time slots. Cells arriving to each input are at once distributed do the CMs, and are stored in COQ related to destined OMs. In the first time slot cells from IP(x, 1) are sent to CM(1), from IP(x, 2) to CM(2), from IP(x, 3) to CM(3); in the second time slot cells from IP(x, 1) are sent to CM(2), from IP(x, 2) to CM(3), from IP(x, 3) to CM(4) and so on. Arriving cells are evenly distributed to CMs, to decrease cell delay within the SMM Clos-network switch.

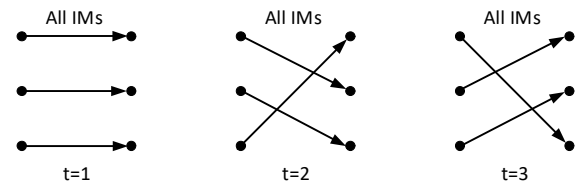

Fig. 2. A sequence in which the static connection patterns should be changed in each IM of capacity 3×3

The SD scheme may be also adopted to the MSM Clos-network switch [7].

III. Input Traffic Analysis

We assume that the traffic directed to each input port IP(*i, h*) can be modeled by i.i.d. Bernoulli process, where the number of successes – which means the number of cells arriving in *t* time slots (in *t* trails) is tp_B with p_B denoting the probability of success in one trial. In such case ports arrival rate is expressed by the expected value:

$$\lambda_{IP} = \lim_{t \to \infty} \frac{t\, p_B}{t} = p_B \tag{1}$$

Therefore, the input traffic arriving to one input module is equal to $\lambda_{IM}=np_B$, and to the whole switching fabric, to all input modules – rnp_B. The SD algorithm balances this input load on CMs and after *m* time slots the central modules arrival rate can be expressed in the following way:

$$\lambda_{CM} = \frac{n\, p_B\, r}{m} \tag{2}$$

There are output queues (COQs) in each central module which store cells destined do the predetermined OMs. Analyzing the input rate of this queues it is easy to see that this rate can be assessed as:

$$\lambda_{COQ(i,j)} = \frac{np_B\, r}{m} p_{ij} \tag{3}$$

where p_{ij} represents the probability that a cell arriving from the *i-th* input module is destined to the *j-th* output module. For example, under the traffic uniformly distributed to the output ports and in consequence to the output modules OMs, $p_{ij}=1/r$. This means that even for maximal input ports load i.e. for

$p_B=1$, the rate $\lambda_{COQ(i,\ j)}$ is less or equal to 1 if the number of OMs is $m \geq n$.

In the investigated SMM Clos-network architecture each central module CM has one link to each OM. This assures that in each time slot from any non-empty COQ*(i, j)* one cell will be sent to the appropriate OM*(j),* which can be described by the COQ*(i, j)* queue's service rate μ=1.

IV. Stabbility Proof

It should be noted that one of the most important characteristics of the switching network under control of a given control algorithm is its throughput and the average and maximum packet delay. Both these parameters depend directly on the stability of such systems.

Intuitively, stability implies that the total number of packets (cells) in the system remains bounded, so that the following equation (4) is satisfied:

$$\lim_{t\to\infty}\frac{q_t}{t}=\lim_{t\to\infty}\frac{1}{t}\sum_{l=1}^{t}(A_l-D_l)=0 \ with\ probab.1 \quad (4)$$

Here, q_t represents the queue-lengths vector at time slot t and D_l and A_l are the departure and arrival vectors at time slot l respectively.

The stability of the switching network means that the length of the queues of cells waiting to be transmitted to output ports does not grow to infinity. This property is extremely important because the length of the queues affects the delay of cells in the system.

The theory of stability for deterministic dynamic systems was founded by A. Lyapunov [10] (see also [11] for survey of stability ideas) who invented two methods for stability investigation. His second method known as *Lyapunov's second method* or *indirect method* turned out to be very effective in proving the stability of very wide spectrum of deterministic systems – linear, non-linear, continuous and discrete. Later, Lyapunov's ideas has been extended on stochastic systems mainly by F. Foster [8]. The application of this theory to Markov chains was done by S. Meyn and R. Tweedie [9]. According to [8] and [9] the stability proof for stochastic systems modeled by Markov chains must show:

- the irreducibility of the chain which means that starting from any initial state it is possible to arrive in subsequent transitions on any other state of the chain;
- the positive recurrence of the chain, which can be done by demonstrating the negative drift of the Lyapunov function.

The function fulfilling Lyapunov conditions can be regarded as Lyapunov candidate function (only the candidate function which allows stability proving is called Lyapunov function). The requirements impose that Lyapunov candidate function $V(\boldsymbol{x})$ [9]:

- is scalar on investigated system's state vector $\boldsymbol{x}$; switching networks' states are determined by queue lengths;
- positive semidefinite, i.e.: $\underset{\boldsymbol{x}\neq 0}{\forall} V(\boldsymbol{x})>0; V(\boldsymbol{0})=0;$
- grows with the state growth of the investigated system which, in our case, means that it grows with the length of switching network queues;
- for continuous systems: $V(x)\in C_1$.

Generally saying, there are two levels of stability [8, 9, 11] - the so-called weak stability and the asymptotic stability. A proof of weak stability for a given switch network guarantees its full, 100% throughput, but does not predetermine the maximum delay of cells, which in general may be unlimited. The asymptotic stability is a more demanding level of stability, which guarantees not only full throughput of the network, but also a finite value of the maximum cells delay.

Formally, the switching system in which the packets (cells) arrival is an independent random process is characterized by the weak (in Lyapunov sense) stochastic stability if for every $\varepsilon > 0$ there exists $\delta > 0$, that:

$$\underset{\varepsilon>0}{\forall}\ \exists\delta>0\ \lim_{t\to\infty}P\{\|q_t\|>\delta\}<\varepsilon \quad (5)$$

or

$$\underset{\varepsilon>0}{\forall}\ \exists\delta>0\ \lim_{t\to\infty}P\{\|q_t\|<\delta\}<1-\varepsilon \quad (6)$$

Where $P\{Z\}$ denotes the probability of the event Z, and $\|q_t\|$ is any norm of q_t – the measure of queues in the system.

The asymptotic stochastic stability is defined as follows: a switching fabric in which the packets (cells) arrival is an independent random stationary process is characterized by asymptotic stochastic stability if:

$$\sup E\{\|q_t\|\}<\infty \quad (7)$$

Inequality (7) means that the maximum expected value of $\|q_t\|$ is finite. The asymptotic stochastic stability guarantees limited average queue lengths and limited cell delay times.

As shown above, the dynamics of the SMM switching fabric is determined by the COQ queues (due to static connections of the central stage with the first and third stages the contentions are possible only in the COQ queues).

Let us note that the dynamics of the COQ(*i, j*) queue can be represented by the Markov chain's state diagram depicted in Fig. 3, where λ represents queue arrival rate - $\lambda_{COQ\ (i,\ j)}$, and μ - is the queue service rate.

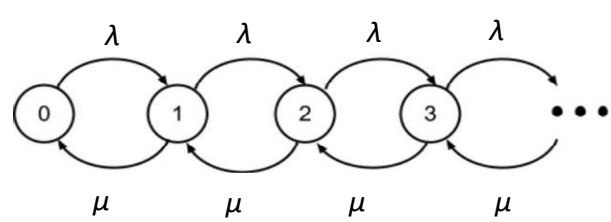

Fig. 3. State graph of the COQ(*i, j*) queue

The proof of stability of this queue can be based on the second Lyapunov method [8, 9, 10]. It requires that Lyapunov

candidate function $V(q_t)$, defined on the queue length, has a negative drift, strictly that:

$$\underset{\|q_t\|>\varepsilon}{\forall} \quad E\left[V(q_{t+1})-V(q_t)\middle|q_t\right]<-\delta \tag{8}$$

In the following proof of stability, Lyapunov candidate function is chosen as the simplest possible one:

$$V(q_t)=q_t \tag{9}$$

The selected function $V(q_t)$ satisfies the previously specified Lyapunov candidate function requirements. After substituting the selected function $V(q_t)$ into the left-hand side of inequality (8) and taking into account the graph in Fig.3:

$$E\left[V(q_{t+1})-V(q_t)\middle|q_t\right]=E[q_{t+1}|q_t]-E[q_t|q_t]=$$
$$=E[q_{t+1}|q_t]-q_t=$$
$$\left[\frac{\lambda}{\lambda+\mu}(q_t+1)+\frac{\mu}{\lambda+\mu}(q_t-1)\right]-q_t=\frac{\lambda-\mu}{\lambda+\mu} \tag{10}$$

Eventually, the stability condition is:

$$\frac{\lambda-\mu}{\lambda+\mu}<0 \tag{11}$$

The drift is negative when $\lambda<\mu$. For $\mu=1$ the system will be weakly stable (stable in Lyapunov sense) for $\lambda<1$. It is worth noting that it does not follow that for $\lambda=1$ the system will not be stable. The Lyapunov method proves only the stability, and if that fails, the instability of the studied system not follows from it.

For proving the asymptotic stochastic stability, it should be shown that:

$$\underset{\|q_t\|>\varepsilon}{\forall} E\left[V(q_{t+1})-V(q_t)\middle|q_t\right]<-\delta\|q_t\| \tag{12}$$

For this purpose, we need another Lyapunov candidate function $V(q_t)$ – we choose it as:

$$V(q_t)=q_t^2 \tag{13}$$

The drift of this function is:

$$E\left[V(q_{t+1})-V(q_t)\middle|q_t\right]=E[q_{t+1}^2|q_t]-E[q_t^2|q_t]$$
$$=E[q_{t+1}^2|q_t]-q_t^2=$$
$$=\left[\frac{\lambda}{\lambda+\mu}(q_t+1)^2+\frac{\mu}{\lambda+\mu}(q_t-1)^2\right]-q_t^2=$$
$$=\frac{\lambda}{\lambda+\mu}\left(q_t^2+2q_t+1\right)+\frac{\mu}{\lambda+\mu}\left(q_t^2-2q_t+1\right)-q_t^2$$
$$=q_t^2\left(\frac{\lambda}{\lambda+\mu}+\frac{\mu}{\lambda+\mu}-1\right)+2q_t\left(\frac{\lambda}{\lambda+\mu}-\frac{\mu}{\lambda+\mu}\right)+$$
$$\frac{\lambda+\mu}{\lambda+\mu}=2q_n\frac{\lambda-\mu}{\lambda+\mu}+1=2q_n\frac{-(\mu-\lambda)}{\lambda+\mu}+1 \tag{14}$$

Solving the inequality:

$$2q_t\frac{-(\mu-\lambda)}{\lambda+\mu}+1<0 \tag{15}$$

the conditions for asymptotic stability can be determined. For $\mu=1$ we obtain:

$$q_t>\frac{\lambda+1}{2(1-\lambda)} \text{ and } \lambda<1 \tag{16}$$

This means that the asymptotic stability will only occur for q_t sufficiently large, for example assuming $\lambda=0.9$, this will be an average of 10 cells, that is, when the value is reached, the cell delay will be limited and stabilized.

V. Simulation Experiments

The experiments have been carried out for the SMM Clos-network switch $C(8, 8, 8)$ of size 64 × 64 (8 switches in each stage) under the SD algorithm. A wide range of traffic load per input port, from $p_B = 0{,}05$ to $p_B = 1$, with the step 0.05, was considered in each simulation experiment. The 95% confidence intervals that have been calculated after *t*-student distribution for ten series with 250 000 time slots (after the starting phase comprising 50 000 time slots, which enables to reach the stable state of the SMM Clos-network switch) are at least one order lower than the mean value of the simulation results, therefore they are not shown in the figures. It is assumed that in the second and third stages the switches with output buffers are used, and the size of buffers is not limited. Three main performance measures have been evaluated: average cell delay in time slots, maximum size of OQs, and throughput. A switch can achieve 100% throughput under the uniform or nonuniform traffic, if the switch is stable, as was defined in [12]. It means that the cell queues do not grow without the limit.

Two packet arrival models are considered in simulation experiments: the Bernoulli arrival model, and the bursty traffic model, where the average burst length is set to 16 cells. Several traffic distribution models (the most popular in this research area) have been considered, which determine the probability p_{ij} that a cell, which arrives at an input i, will be directed to an output j. The considered cell distribution models are: uniform - $p_{ij} = p_B/N$, diagonal - $p_{ij} = 2p_B/3$ for $i = j$ and $p_{ij} = p_B/3$ for $j = (i+1) \bmod N$, and 0 otherwise, and Hot-spot: $p_{ij} = p_B/2$ for $i = j$, and $p_B/2(N-1)$ for $i \neq j$.

Selected simulation results are shown in Fig. 4, and 5. Fig. 4 shows average cell delay, in time slots, obtained for Bernoulli and bursty arrival models, and different kind of cell distribution models. The SD algorithm provides 100% throughput for the investigated switching fabric only for uniform traffic and Bernoulli arrival model. Under Bernoulli arrivals, the throughput is limited to 90% for non-uniform traffic, like diagonal and Hot-spot. It is possible to say, that the SD scheme, for the uniform and non-uniform traffic distribution patterns under Bernoulli arrivals, performs quite

well, when the input load is smaller than 0.85. In this case, the average cell delay is not greater than 10 time slots. For the bursty arrival model the SMM Clos-network switch controlled by the SD algorithm is not able to achieve the 100% throughput for both the uniform and nonuniform traffic distribution patterns. For the uniform traffic, the throughput is close to 98%, but for the non-uniform traffic the throughput is limited to 80%.

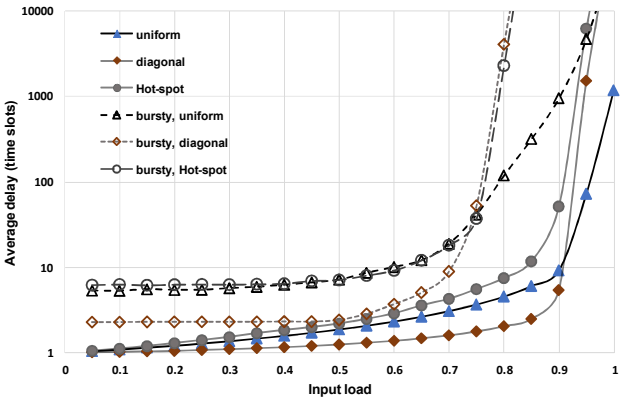

Fig. 4. Average cell delay at egress side of the SMM Clos-network switch under the SD scheme

Fig. 5 shows maximum OQ length obtained during simulation experiments. These results are consistent with charts presented in Fig. 4. It can be seen that for Bernoulli arrivals the OQ length rapidly grow for heavy input load and non-uniform traffic (p_B>0.9). For the bursty traffic the OQ length increasing very fast for p_B>0.75, especially for non-uniform cell distribution patterns.

Generally saying, the SD algorithm is very simple in implementation within the SMM Clos-network switch and can produce good results for input load p_B<0.7 for both the uniform and nonuniform traffic distribution patterns. The results related to the throughput are not very good, but the complexity of this algorithm is very low.

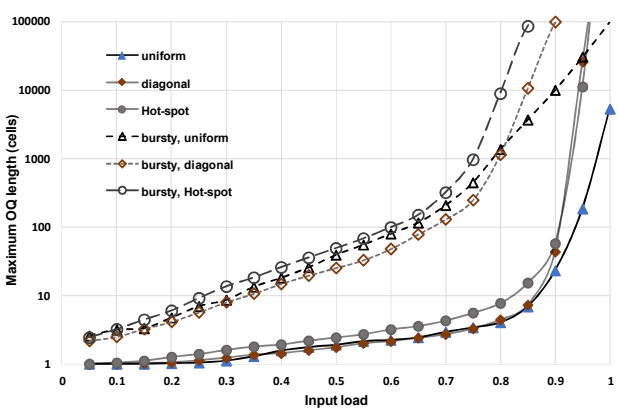

Fig. 5. Maksimum OQ length in OMs under the SD scheme

VI. Conclusions

This paper aims at performance evaluation of the SMM Clos-network switch under the packet dispatching scheme employing static connection patterns, called SD. The system is evaluated in terms of stability and basic performance measures like average cell delay and packet queue lengths. In Section IV we show how it is possible to use the second Lyapunow method to prove the stability of the SMM Clos-network switch under the SD algorithm. Taking into account, that the stability is proven for ideal, theoretical traffic, in Section V we show simulation results obtained for uniform and non-uniform traffic distribution patterns, and Bernoulli and bursty arrival models. The investigated cell dispatching scheme is very simple, but it is not able to provide good performance of the SMM Clos-network switch for very high input load (p_B>0.7), especially for bursty traffic. It is also not possible to provide in-sequence service under this algorithm, which results in special resequencing buffers at outputs.

Acknowledgment

The work described in this paper was financed from the funds of the Ministry of Science and Higher Education for the year 2017 under Grants 08/82/DSPB/8221 and 04/45/DSPB/0162.

References

[1] H. J. Chao and B. Liu, High Performance Switches and Routers. Wiley-Interscience, A John Wiley & Sons, US: New Jersey, 2007.

[2] C. Clos, "A Study of Non-Blocking Switching Networks", *Bell Sys. Tech. Jour*., 1953, pp. 406-424.

[3] Router-Switch.com: Cisco CRS-X Core Router to Offer 10 Times Capacity of Original, http://blog.router-switch.com/2013/06/cisco-crs-x-core-router-to-offer-10-times-capacity-of-original/.

[4] X. Li., Z. Zhou, and M. Hamdi, "Space-Memory-Memory architecture for Clos-network packet switches". *Proc. IEEE International Conference on Communications – ICC 2005*, vol. 2., pp. 1031 – 1035.

[5] K. Yoshigoe, "The Crosspoint-Queued Switches with Virtual Crosspoint Queueing". *Proc. 5th International Conference on Signal Processing and Communication Systems*, ICSPCS 2011, pp. 277-281.

[6] J. Kleban and U. Suszyńska, "Static Dispatching with Internal Backpressure Scheme for SMM Clos-Network Switches", *Proc. The Eighteenth IEEE Symposium on Computers and Communications, ISCC'13*, Split, Croatia, 2013.

[7] J. Kleban and H. Santos, "Packet Dispatching Algorithms with the Static Connection Patterns Scheme for Three-Stage Buffered Clos-Network Switches", *Proc. IEEE International Conference on Communications 2007 - ICC-2007*, Glasgow, Scotland, 2013.

[8] F. G. Foster, "On the stochastic matrices associated with certain queuing processes". *Ann. Math. Statistics.* 24, 1953, pp. 355-360.

[9] S. Meyn and R. Tweedie, Markov Chains and Stochastic Stability. Springer, New York, 1993.

[10] A. M. Lyapunov, "*The General Problem of the Stability of Motion"* (In Russian), Doctoral dissertation, Univ. Kharkov. English translations: (1) *Stability of Motion*, Academic Press, New-York & London, 1966 (2) *The General Problem of the Stability of Motion*, Taylor & Francis, London 1992.

[11] J. Kleban and J. Warczyński, (in Polish): Stabilność buforowanych pól komutacyjnych Closa. Przegląd Telekomunikacyjny, Rocznik LXXXIX and Wiadomości Telekomunikacyjne, Rocznik LXXXV, nr 8-9, 2016, str. 976-981. XXXII Krajowe Sympozjum Telekomunikacji i Teleinformatyki KSTiT'2016. Gliwice 26-28.09.2016.

[12] N. McKeown, A. Mekkittikul, V. Anantharam, and J. Walrand, "Achieving 100% Throughput in an Input-queued Switch", *IEEE Trans. Commun.,* Aug. 1999, pp. 1260-1267.