Modelling and performance evaluation of a novel internal priority routing scheme for finite-buffered multistage interconnection networks

D. C. Vasiliadis[a,b,*] , G. E. Rizos[a,b] , C. Vassilakis [a] and E. Glavas [b]

[a] *Department of Computer Science and Technology, University of Peloponnese, Tripolis, Greece*

[b] *Technological Educational Institute of Epirus, Arta, Greece*

[*] Corresponding author: Dimitris Vasiliadis; tel.: +30 26810 50505; fax: +30 26810 76941 e-mail: dvas@uop.gr ; dvas@teiep.gr .

Dr. Dimitris Vasiliadis is currently head of the Network Operations Center (NOC) of Technological Educational Institute (T.E.I.) of Epirus, a node of the Greek Research and Technology Network (GRNET). He has published more than 30 papers. He has been a PC member and referee in various international journals and conferences. His research interests include Performance Analysis of Networking and Computer Systems, Computer Networks and Protocols, Security of Networks, Telematics, QoS and New Services.

Dr. George Rizos has recently received his PhD degree from the University of Peloponnese at Greece. He is currently a System Administrator of the Network Operations Center (NOC) of the Technological Educational Institute (T.E.I.) of Epirus. He has published several papers. His research interests are focused on Wireless Communications, Ad hoc Networks and Protocols, Multimedia, and Quality of Networking Services.

Dr. Costas Vassilakis is currently an Associate Professor in the Department of Computer Science and Technology, University of Peloponnese, Greece. He has published more than 100 papers. He has been a PC member and referee in several international journals and conferences. His research interests include Semantic Web Technologies and Applications, Service-oriented Architectures and Distributed Systems, E-government and E-commerce, Temporal Database Management Systems, VR-Systems and Visualization, Object Oriented Programming, Computer Networks and Protocols.

Dr. Euripidis Glavas received his Ph.D. degree from the University of Sussex in 1988. He is currently a professor at the Department of Informatics and Telecommunications of the Technological Educational Institute (T.E.I.) of Epirus. He has published many papers and he has been a Program Committee member and referee in various international journals and conferences. His research areas include Optical Wave Guide, Computer Architecture, Parallel and Distributed Systems, and Neural Networks.

Modelling and performance evaluation of a novel internal priority routing scheme for finite-buffered multistage interconnection networks

In this paper, the modelling, analysis and performance evaluation of a novel architecture for internal priority finite-buffered Multistage Interconnection Networks (MINs) is presented. We model the proposed architecture giving the details of its operation and describing its states and detailing conditions and effects of state transition; we also provide a formal model for evaluating its performance. The proposed architecture's performance is subsequently analyzed under the uniform traffic condition, considering various offered loads, buffer-lengths and MIN sizes, using simulations. We compare the internal priority scheme vs. the non priority (or single priority) scheme, by gathering metrics for the two most important network performance factors, namely packet throughput and the mean time a packet needs to traverse the network. We demonstrate and quantify the improvements on MIN performance stemming from the introduction of priorities in terms of throughput and a combined performance indicator which depicts the overall performance of the MIN. These performance measures can be valuable assets for designers of parallel multiprocessor systems and networks in order to minimize the overall deployment costs and delivering efficient systems.

Keywords: Multistage Interconnection Networks; Banyan Switches; Packet Switching; Performance Analysis; Simulation Model

Introduction

Multistage Interconnection Networks (MINs) with crossbar Switching Elements (SEs) are frequently proposed for interconnecting processors and memory modules in parallel multiprocessor systems [1], [2], [3]. MINs have been recently identified as an efficient interconnection network for communication structures such as gigabit Ethernet switches, terabit routers, and ATM switches [4], [5], [6]. Significant advantages of MINs include their low cost/performance ratio and their ability to route multiple communication tasks concurrently. MINs with the Banyan [7] property are proposed to connect a large number of processors to establish a multiprocessor system; they have also received considerable interest in the development of packet-switched networks. Non-Banyan MINs, are in general, more expensive than Banyan networks and more complex to control.

In a parallel or distributed system, the performance of the network interconnecting the constituent elements (nodes, processors, memory modules etc) is a critical factor for the overall system performance. Much research has been therefore conducted during the last decades in the area of investigating the performance of networks and communications facilities. In order to evaluate network performance different methods have been used, mainly classified in two major categories. The first category includes analytical models based either on Markov models or on Petri-nets, whereas the second category employs simulation to estimate network performance. Accurate performance estimation before network implementation is of essence, since it allows network designers to adapt network design and tune operational parameters to the specific requirements of the system under implementation, enabling thus building of efficient systems, cost reduction and minimization of rollout times.

In this paper we propose a novel two-level internal priority scheme for performing routing within the MIN. The proposed scheme takes into account the queue lengths of the MIN switching elements, prioritizing packets in SEs having greater queue lengths. The rationale behind this approach is that by offloading large queues, the probability that buffers fill up decreases, thus less packets will be dropped due to buffer shortage. This is expected to increase network performance, while fairness between packets is also promoted. The performance of the proposed scheme is also evaluated and compared against that of single-priority MINs.

The remainder of this paper is organized as follows: section 2 overviews related work in the area of network performance evaluation and priority schemes, while in section 3 we present the proposed priority scheme -which is termed as internal priority- and give an analytical model for finite-buffered MINs with internal priority and non priority scheme SEs. The analytical model employs a novel 5-states buffer model. Subsequently, in section 4 we present the performance criteria and parameters related to the network. Section 5 presents the results of our performance analysis, which has been conducted through simulation experiments, while section 6 provides the concluding remarks and outlines future work.

Related work

The principal methods for estimating network performance are analytical modelling and simulation. Markov chains, which fall in the analytical modelling category, have been extensively used by many researchers. In [8] and [9] Markov chains are used in order to approximate the behaviour of MINs under different buffering schemes. In [8], particularly, Markov chains are enhanced with elements from queuing theory. Petri nets [10] [11] [12] have been also used as modelling methods either to complement Markov chains or as self-contained approaches. The studies reported in [13] and [14] studied MINs with uniform load traffic on inputs. Hot-spot traffic performance was also examined by Jurczyk [15], while Turner [16] dealt with multicast in Clos networks, as a subclass of MINs. Atiquzzaman [17] focused only on non-uniform arriving traffic schemes. Furthermore, Kleinrock [18] discusses approaches that examine the case of Poisson traffic on inputs of a MIN. In the industry domain, Cisco has built its new CRS-1 router [19] [20] as a multistage switching fabric. The switching fabric that provides the communications path between line cards is a 3-stage, self-routed architecture.

Packet priority is a common issue in networks, arising when some packets need to be offered better quality of service than others. Packets with real-time requirements (e.g. from streaming media) vs. non real-time packets (e.g. file transfer), and out-of-band data vs. ordinary TCP traffic [21] are two examples of such differentiations. There are already several commercial switches which accommodate traffic priority schemes, such as [22] [23]. These switches consist internally of single priority SEs and employ two priority queues for each input port, where packets are queued based on their priority level. Chen and Guerin [24] studied an (N X N) non-blocking packet switch with input queues, built using one-priority SEs. Ng and Dewar [25] introduced a simple modification to load-sharing replicated buffered Banyan networks to guarantee priority traffic transmission.

In this paper, a different type of packet priority is employed. Contrary to other approaches [36] [37], where priority is defined at the application layer (e.g. real-time packets from streaming media vs. non real-time packets from file transfer; out-of-band data vs. ordinary TCP traffic [26] and so forth) or in the parallel systems architecture level (e.g. processor-memory traffic regarding operating system operations is prioritized against user process' traffic) in the proposed architecture packet priority is computed dynamically and is directly proportional to the transmission queue length of the SE that the packet is currently stored in. This priority is used for resolving buffer contentions, which in typical MINs are resolved by randomly dropping one of the contending packets.

Internal priority MIN and analytical MIN model

A MIN can be defined as a network used to interconnect a group of *N* inputs to a group of *M* outputs using several stages of small size *switching elements* (SEs) followed (or leaded) by link states. It is usually defined by, among others, its topology, routing algorithm, switching strategy and flow control mechanism. A MIN with the Banyan property is defined in [7] and is characterized by the fact that there is exactly one unique path from each source (input) to each sink (output). Banyan MINs are multistage self-routing switching fabrics. Thus, each SE of k^{th} stage can decide in which output port to route a packet to, depending on the corresponding k^{th} bit of the destination address.

An (N X N) MIN can be constructed by n=log$_c N$ stages of (cxc) SEs, where c is the degree of the SEs. A typical SE is illustrated in fig. 1. At each stage there are exactly N/c SEs, consequently the total number of SEs of a MIN is (N/c)∗log$_c N$. Thus, there are O(N*log$N$) interconnections among all stages, as opposed to the crossbar network which requires O(N^2) links.

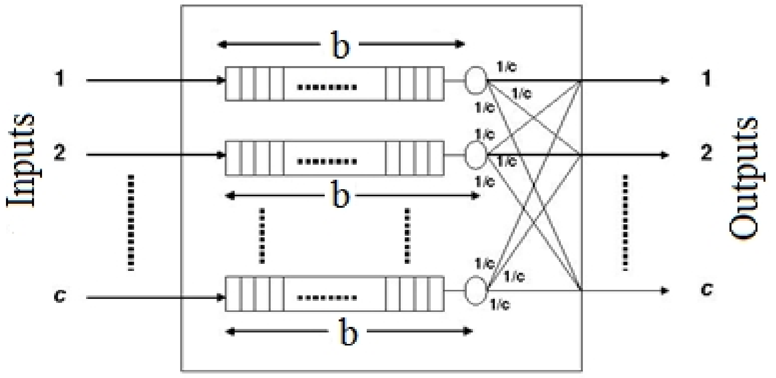

Fig. 1. An cxc Switching Element

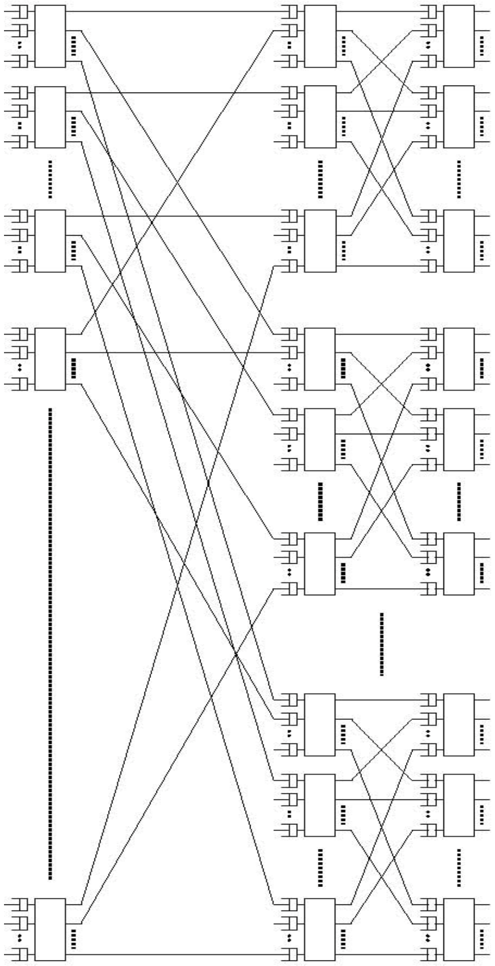

Fig. 2. 3-stage Delta Network consisting of cxc SEs

A typical configuration of an *N* X *N* delta network, one of the most widely used classes of Banyan MINs, which were proposed by Patel [29], is shown at fig. 2.

In our paper, we consider a Multistage Interconnection Network with the Banyan property that operates under the following assumptions:

- The network clock cycle consists of two phases. In the first phase, flow control information passes through the network from the last stage to the first one. Flow control information generally includes data regarding the size of the queues in the subsequent stages and congestion control tags [38] [39]. In the second phase, packets flow from one stage to the next in accordance to the flow control information. SEs operate in a slotted time model [14] and routing is performed in a pipeline manner, meaning that the routing process occurs in every stage in parallel.
- The arrival process of each input of the network is a simple Bernoulli process, i.e. the probability that a packet arrives within a clock cycle is constant and the arrivals are independent of each other. We will denote this probability as λ.
- A packet arriving at the first stage (k=1) is discarded if the buffer of the corresponding SE is full.
- All SEs have deterministic service time.

- A packet is blocked at a stage if the destination buffer at the next stage is full.
- The packets are uniformly distributed across all the destinations and each queue uses a FIFO policy for all output ports.
- When two packets at a stage contend for a buffer at the next stage and there is not adequate free space for both of them to be stored (i.e. only one buffer position is available at the next stage), there is a conflict. In the single priority (or no-priority) scheme MINs, one packet will be accepted at random and the other will be blocked by means of upstream control signals. In the proposed internal-priority scheme, if a conflict occurs it is resolved by examining the number of packets within the transmission queue of the SEs from which the contending packets originate. For such a decision, however, to be taken, the receiving SE needs to have available the queue lengths of the transmitting SEs, a piece of information which is not available to the receiving SE in typical MINs. To make this information available, SEs operating under the internal priority MIN scheme send the length of their transmission packet queue at the start of the packet header, as a preamble. When receiving SEs detect a conflict situation (i.e. two incoming transmissions and only one free buffer slot), they compare the queue sizes of the transmitting SEs and proceed in receiving the packet preambled with the largest value for the queue size. The other packet will be blocked, and the transmitting SE will be notified by means of an upstream control signal during the next network cycle, as in the "typical" MIN operation. Since buffer sizes in SEs are usually in the range 1 to 16, the length of the preamble can vary from 1 to 4 bits (in our study the length of the preamble was set to 3), which is quite small compared to the packet length. The preamble need not be checksumed (which would increase its size), since any error in these bits would (in the worst case) simply lead to accepting the wrong (with respect to the priority policy) packet, a case that would only marginally affect the gains obtained by the introduction of the internal priority scheme.
- Finally, all packets in input ports contain both the data to be transferred and the routing tag. In order to achieve synchronously operating SEs, the MIN is internally clocked. As soon as packets reach a destination port they are removed from the MIN, so, packets cannot be blocked at the last stage.

Our analysis introduces a novel model, which considers not only the current state of the associated buffer, but also the previous one. I.e. in the case of a single-bufferd MIN based on the one clock history consideration we enhance the Mun's [30] three states model with a five states buffer model, which is described in the following paragraphs.

Analysis

Since the proposed model is exemplified in a single-buffered configuration, the buffer state will be either empty '0' or full '1' at each clock cycle. Regarding one clock history consideration we examine the subsequent states:

- State '00': Buffer was empty at the beginning of the previous clock cycle and it is also empty at beginning of the current clock cycle (i.e. no new packet has been received during the previous clock cycle; buffer remains empty).
- *State '01'*: Buffer was empty at the beginning of the previous clock cycle, while it contains a new packet at the current clock cycle (i.e. a new packet has been received during the previous clock cycle; buffer is filled now).
- *State '10'*: Buffer had a packet at the previous clock cycle, while it contains no packet at the current clock cycle (i.e. a packet has been sent during the previous clock cycle, but no new packet has been received; buffer is empty now).
- *State '11n'*: Buffer had a packet at the previous clock cycle and has a new packet at the current clock cycle (i.e. a packet has been sent during the previous clock cycle, and a new packet has also been received; buffer is filled with a new packet now).
- *State '11b'*: Buffer had a packet at the previous clock cycle and has a blocked packet at the current clock cycle (i.e. no packet has been sent during the previous clock cycle due to blocking; buffer is filled with the blocked packet now).

The following variables are defined in order to develop an analytical model. In all definitions SE(k) denotes a SE at *stage k* of the MIN.

Definitions

- $P_{00}(k,t)$ is the probability that a buffer of SE(k) is empty at both $(t\text{-}1)^{th}$ and t^{th} network cycles.
- $P_{01}(k,t)$ is the probability that a buffer of SE(k) is empty at $(t\text{-}1)^{th}$ network cycle and has a new packet at t^{th} network cycle.
- $P_{10}(k,t)$ is the probability that a buffer of SE(k) has a packet at $(t\text{-}1)^{th}$ network cycle and has no packet at t^{th} network cycle.
- $P_{11n}(k,t)$ is the probability that a buffer of SE(k) has a packet at $(t\text{-}1)^{th}$ network cycle and has also a new one at t^{th} network cycle.
- $P_{11b}(k,t)$ is the probability that a buffer of SE(k) has a packet at $(t\text{-}1)^{th}$ network cycle and has a blocked one at t^{th} network cycle.
- $q(k,t)$ is the probability that a packet is ready to be accepted to a buffer of SE(k) at t^{th} network cycle.
- $r_{01}(k,t)$ is the probability that a packet in a buffer of SE(k) is ready to move forward during the t^{th} network cycle, given that the buffer is in '01'state.
- $r_{11n}(k,t)$ is the probability that a packet in a buffer of SE(k) is ready to move forward during the t^{th} network cycle, given that the buffer is in '11n' state.

- $r_{11b}(k,t)$ is the probability that a packet in a buffer of SE(k) is ready to move forward during the t^{th} network cycle, given that the buffer is in '11b' state.

The following equations represent the evolution of the state probabilities as the clock cycles advance. These equations are derived from the state transition diagram at fig. 3.

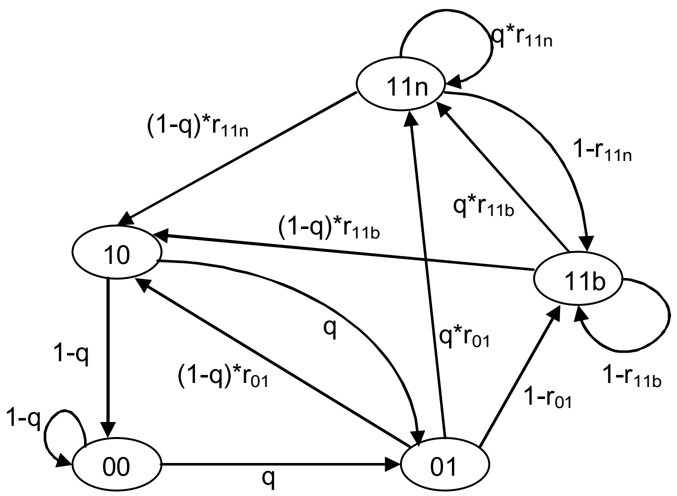

Fig. 3. A state transition diagram of a SE(k) buffer.

The probability that a buffer of SE(k) was empty at the $(t$-1$)^{th}$ network cycle is $P_{00}(k,t$-1) + $P_{10}(k,t$-1). Therefore, the probability that a buffer of SE(k) is empty both at the current t^{th} and previous $(t$-1$)^{th}$ network cycles is the probability that the SE(k) was empty at the previous $(t$-1$)^{th}$ network cycle multiplied by the probability 1 - q(k,t-1) of no packet was ready to be forwarded at the SE(k) during the previous network cycle (the two facts are statistically independent, thus the probability that both are true is equal to the product of the individual probabilities). Formally, this probability $P_{00}(k,t)$ can be expressed by

$$P_{00}(k,t) = [1-q(k,t-1)] * [P_{00}(k,t-1) + P_{10}(k,t-1)] \quad (1)$$

The probability that a buffer of SE(k) was empty at the $(t$-1$)^{th}$ network cycle and a new packet has arrived at the current t^{th} network cycle is the probability that the SE(k) was empty at the $(t$-1$)^{th}$ network cycle [which is equal to $P_{00}(k,t$-1) + $P_{10}(k,t$-1)] multiplied by the probability q(k,t-1) that a new packet was ready to be transmitted to

SE(k) during the (t-1)th network cycle. Formally, this probability $P_{01}(k,t)$ can be expressed by

$$P_{01}(k,t) = q(k,t-1) * [P_{00}(k,t-1) + P_{10}(k,t-1)] \ (2)$$

The case that a buffer of SE(k) was full at the (t-1)th network cycle but is empty during the (t-1)th network cycle effectively requires the following two facts to be true: (a) a buffer of SE(k) was full at the (t-1)th network cycle *and* the packet was successfully transmitted and (b) no packet was received during the (t-1)th network cycle to replace the transmitted packet into the buffer. The probability for fact (a) is equal to $r_{01}(k,t\text{-}1) * P_{01}(k,t\text{-}1) + r_{11n}(k,t\text{-}1) * P_{11n}(k,t\text{-}1) + r_{11b}(k,t\text{-}1) * P_{11b}(k,t\text{-}1)$; this is computed by considering all cases that during the network cycle *t-1* the SE had a packet in its buffer and multiplying the probability of each state by the corresponding probability that the packet was successfully transmitted. The probability of fact (b), i.e. that no packet was ready to be transmitted to SE(k) during the previous network cycle is equal to 1 - $q(k,t\text{-}1)$. Formally, the probability $P_{10}(k,t)$ can be computed by the following formula:

$$P_{10}(k,t) = [1-q(k,t-1)] * [r_{01}(k,t-1) * P_{01}(k,t-1) + r_{11n}(k,t-1) * P_{11n}(k,t-1) + r_{11b}(k,t-1) * P_{11b}(k,t-1)] \ (3)$$

The probability that a buffer of SE(k) had a packet at the (t-1)th network cycle and has also a new one (different than the previous; the case of having the same packet in the buffer is addressed in the next paragraph) at the t^{th} network cycle is the probability of having a ready packet to move forward at the previous (t-1)th network cycle [which is equal to $r_{01}(k,t\text{-}1) * P_{01}(k,t\text{-}1) + r_{11n}(k,t\text{-}1) * P_{11n}(k,t\text{-}1) + r_{11b}(k,t\text{-}1) * P_{11b}(k,t\text{-}1)$] multiplied by $q(k,t\text{-}1)$, i.e. the probability that a packet was ready to be transmitted to SE(k) during the previous network cycle. Formally, this probability $P_{11n}(k,t)$ can be expressed by

$$P_{11n}(k,t)=q(k,t-1) * [r_{01}(k,t-1) * P_{01}(k,t-1)+r_{11n}(k,t-1) * P_{11n}(k,t-1)+ r_{11b}(k,t-1) * P_{11b}(k,t-1)] \quad (4)$$

The final case that should be considered is when a buffer of SE(*k*) had a packet at the (*t*-1)th network cycle and still contains *the same packet* at the *t*th network cycle. This occurs when the packet in the buffer of SE(*k*) was ready to move forward at the (*t*-1)th network cycle, but it was blocked (not forwarded) during that cycle, due to a blocking event -either the associated buffer of the next stage SE was already filled due to another blocking, or it was occupied by a second packet of the current stage contending for the same buffer during the process of forwarding. The probability for this case can be formally defined as

$$P_{11b}(k,t) = [1-r_{01}(k,t-1)] * P_{01}(k,t-1) + [1-r_{11n}(k,t-1)] * P_{11n}(k,t-1)+[1-r_{11b}(k,t-1)] * P_{11b}(k,t-1) \quad (5)$$

Adding the equations (1) ... (5), both left and right-hand sides are equal to 1 validating thus that all possible cases have been covered; indeed, P00(k,t) + P01(k,t) + P10(k,t) + P11n(k,t) + P11b(k,t) = 1 and P00(k,t-1) + P01(k,t-1) + P10(k,t-1) + P11n(k,t-1) + P11b(k,t-1) = 1. The system of equations presented in the previous paragraphs extends the ones presented in other works (e.g. [11], [13], [14], [31]) by considering the state and transitions occurring within an additional clock cycle. All previous works were based on a three states model. This enhancement with a five states buffer model can improve the accuracy of the performance parameters calculation (*throughput* and *delay*). The simulation presented in section 5 takes into account all the above presented dependencies among the queues of each SE(*k*) of the MIN. As compared with the work presented in [31], the accuracy of both throughput and delay calculation has increased by 3% (in [31] the results diverted up to 4% from Theimer's model [14] which is considered to be the most accurate, while in the work presented here the diversion margin has dropped to 1%). In our future work, we aim to study in detail an analytical model for double-buffered MINs, incorporating the internal priority scheme and validate the analytical model through simulations. Part of this future work is the analytic computation of the probabilities listed in the Definitions section above; currently, all these probabilities are computed through simulation.

Performance Evaluation Methodology

In order to evaluate the performance of a (N X N) MIN with n=log$_c N$ intermediate stages of (cxc) SEs, we use the following metrics. Let T be a relatively large time period divided into u discrete time intervals ($\tau_1, \tau_2, \dots, \tau_u$).

- *Average throughput* Θ_{avg} is the average number of packets accepted by all destinations per network cycle. This metric is also referred to as *bandwidth*. Formally, Θ_{avg} can be defined as

$$\Theta_{avg} = \lim_{u \to \infty} \frac{\sum_{i=1}^{u} n(i)}{u} \quad (6)$$

where $n(i)$ denotes the number of packets that reach their destinations during the i^{th} time interval.

- *Normalized throughput* Θ is the ratio of the *average throughput* Θ_{avg} to network size N. Formally, Θ can be expressed by

$$\Theta = \frac{\Theta_{avg}}{N} \quad (7)$$

Normalized throughput is a good metric for assessing the MIN's cost effectiveness.

- *Average packet delay* D_{avg} is the average time a packet spends to pass through the network. Formally, D_{avg} can expressed by

$$D_{avg} = \lim_{u \to \infty} \frac{\sum_{i=1}^{n(u)} t_d(i)}{n(u)} \quad (8)$$

where $n(u)$ denotes the total number of packets accepted within u time intervals and $td(i)$ represents the total delay for the i^{th} packet. We consider $t_d(i)$=$t_w(i)$ + $t_{tr}(i)$ where $t_w(i)$ denotes the total queuing delay for i^{th} packet waiting at each stage for the availability of an empty buffer at the next stage queue of the network. The second term $t_{tr}(i)$ denotes the total transmission delay for i^{th} packet at each stage of the network; this is equal to $n*nc$, where n is the number of stages and nc is the network cycle.

- *Normalized packet delay* D is the ratio of the D_{avg} to the minimum packet delay which is simply the transmission delay $n*nc$. Formally, D can be defined as

$$D = \frac{D_{avg}}{n * nc} \quad (9)$$

- *Universal performance (U)* is defined by a relation involving two above normalized factors, D and Θ: A MIN's performance is considered optimal when D is minimized and Θ is maximized, thus the formula for computing the universal factor arranges so that the overall performance metric for a MIN follows this rule. Formally , U can be expressed by

$$U = \sqrt{D^2 + \frac{1}{\Theta^2}} \quad (10)$$

It is obvious that, when the packet delay factor becomes smaller or/and throughput factor becomes larger the universal performance factor (U) becomes smaller. Consequently, as the universal performance factor (U) becomes smaller, the performance of a MIN is considered to improve. Because the above factors (parameters) have different measurement units and scaling, we normalize them to obtain a reference value domain. Normalization is performed by dividing the value of each factor by the (algebraic) minimum or maximum value that this factor may attain. Thus, equation (10) can be replaced by:

$$U = \sqrt{\left(\frac{D - D^{\min}}{D^{\min}}\right)^2 + \left(\frac{\Theta^{\max} - \Theta}{\Theta}\right)^2} \quad (11)$$

where $D^{\min}$ is the minimum value of normalized packet delay (D) and $\Theta^{\max}$ is the maximum value of normalized throughput. Consistently to equation (10), when the universal performance factor U, as computed by equation 11 is close to zero, the MIN performance is considered optimal whereas, when the value of U increases, the MIN performance deteriorates. Finally, taking into account that the values of both delay and throughput appearing in equation (11) are normalized, $D^{\min} = \Theta^{\max} = 1$, thus the equation can be simplified to:

$$U = \sqrt{(D-1)^2 + \left(\frac{1-\Theta}{\Theta}\right)^2} \quad (12)$$

Finally, we list the major parameters affecting the performance of a MIN.

- *Buffer size* (b) is the maximum number of packets that an input buffer of a SE can hold. In our paper we consider a finite-buffered (b = 1, 2, 4, 8) MIN.

- *Probability of arrivals* (λ) is the steady-state fixed probability of arriving packets at each queue on inputs. In our simulation λ is assumed to be $\lambda = 0.1$, 0.2, … , 0.9, 1.

- *Number of stages n*, where $n = \log_2 N$, is the number of stages of an (N X N) MIN. In our simulation n is assumed to be $n = 3, 6, 8, 10$.

Simulation and performance results

The performance of MINs is usually determined by modelling, using simulation [31] [32] or mathematical methods [33] [35]. In this paper we estimated the network performance using simulations. We developed a generic simulator for MINs in a packet communication environment. The simulator can handle several switch types, inter-stage interconnection patterns, load conditions, switch operation policies, and priorities. We focused on an (*N* X *N*) Delta Network that consists of (2 X 2) SEs, using internal queuing. Each (2 X 2) SE in all stages of the MIN was modelled by two non-shared buffer queues. Buffer operation was based on FCFS principle. In the case of non-priority scheme MINs, when there was a contention between two packets, it was solved randomly. The performance of non-priority MINs was compared against the performance of internal priority MINs, where contentions were resolved by favouring the packet transmitted from the SE with the highest transmission queue length. The simulation was performed at packet level, assuming fixed-length packets transmitted in equal-length time slots, where the slot was the time required to forward a packet from one stage to the next.

The parameters for the packet traffic model were varied across simulation experiments to generate different offered loads and traffic patterns. Metrics such as packet throughput and packet delays were collected at the output ports. We performed extensive simulations to validate our results. All statistics obtained from simulation running for 10^5 clock cycles. The number of simulation runs was adjusted to ensure a steady-state operating condition for the MIN. There was a stabilization process in order that the network would be allowed to reach a steady state by discarding the first 10^3 network cycles, before collecting the statistics.

Fig. 4 shows the *normalized throughput* of a single-buffered MIN with 6 stages as a function of the *probability of arrivals* for the three classical models [14] [30] [34] and our simulation. All models are very accurate at low loads. The accuracy reduces as input load increases. Especially, when input load approaches the network maximum throughput, the accuracy of Jenq's model is insufficient. One of the reasons is the fact that many packets are blocked mainly at the network first stages at high traffic rates. Thus, Mun introduced a "blocked" state to his model to improve accuracy. Theimer's model considers the dependencies between the two buffers of an SE; this has lead to further improvement in accuracy and therefore Theimer's model is considered the most accurate insofar. Our simulation was also tested by comparing the results of the Theimer's model with those of our simulation experiments, which were found to be in close agreement (differences are less than 1%).

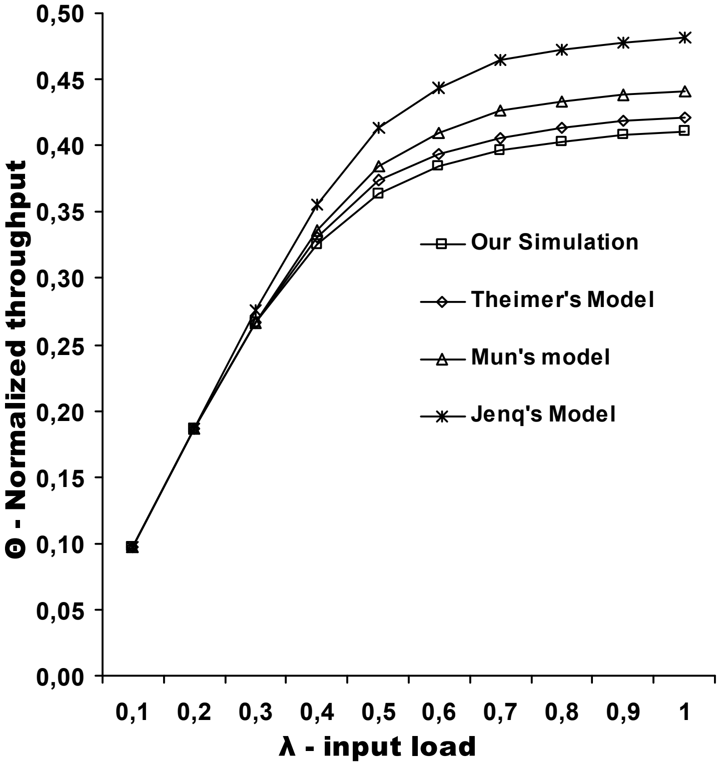

Fig. 4 Normalized throughput of a single buffered 6-stage MIN

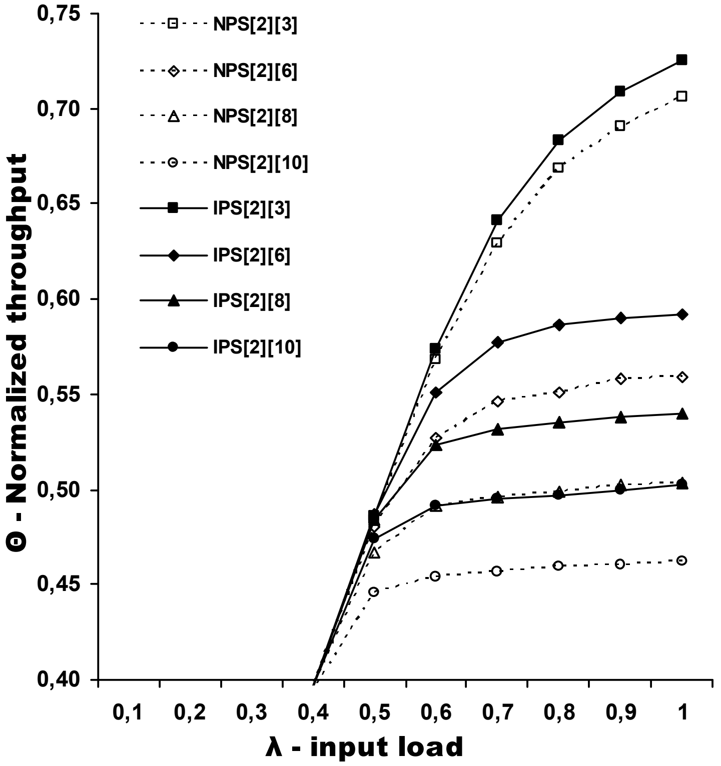

Fig. 5. Normalized throughput of a double buffered *n*-stage (*n*=3,6,8,10) MIN

Fig. 5 illustrates the gains on *normalized throughput* of a MIN using an internal priority vs. non priority (or single priority) scheme. In the diagram, curve NPS[*b*][*n*] depicts the *normalized throughput* of an *n*-stage MIN constructed by 2X2 SEs, using queues *of buffer-length b*, employing a non priority scheme. Similarly, curve IPS[*b*][*n*] shows the corresponding *normalized throughput of* an *n*-stage MIN constructed by 2X2 SEs, using queues of *buffer-length b*, employing an internal priority scheme. In this figure, all curves represent the performance factor of *normalized throughput* for double buffered MINs (*b*=2) at different offered loads (λ=0.1, 0.2, …, 1). We can notice here that the gains on *normalized throughput* of a MIN using an internal priority vs. non priority scheme are 1.9%, 3.3%, 3.7%, and 4.0%, when n=3, 6, 8, and 10 respectively, under full load traffic. It is obvious that the *normalized throughput* falls as the *network size (bandwidth)* increases. However, the gains of *normalized throughput* using the internal priority vs. non priority scheme are more considerable as the *network size* increases.

Fig. 6 illustrates the gains on *normalized throughput* of a MIN using an internal priority scheme as compared to the single priority one in the case of *buffer size* b=4. We can notice here that the gains on *normalized throughput* of a MIN using an internal priority vs. non priority scheme are 1.4%, 3.7%, 4.3%, and 4,7%, when n=3, 6, 8, and 10 respectively, under full load traffic. As it is seen by the diagram the gains on *normalized throughput* remain considerable for all network setups, especially in cases where n>=6.

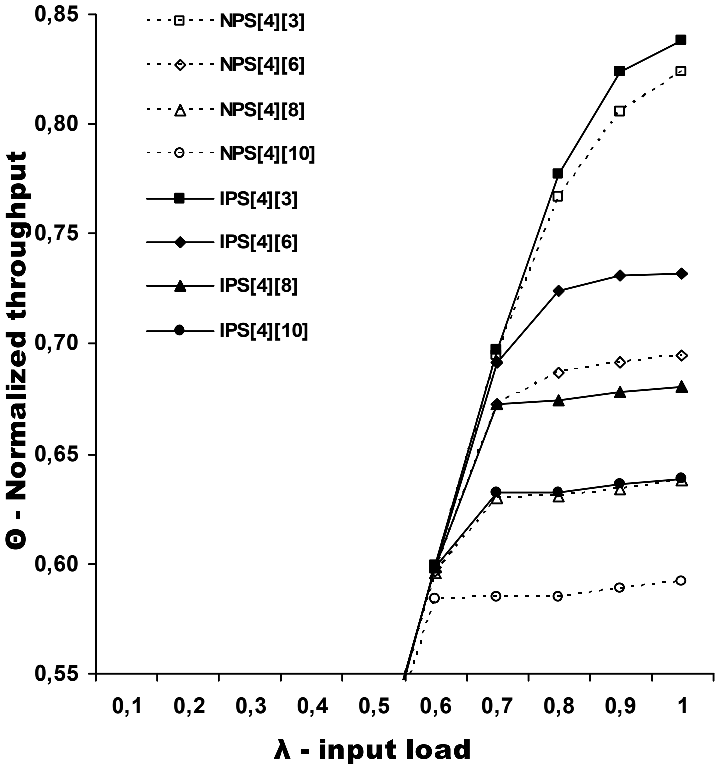

Fig. 6. Normalized throughput of a finite-buffered (b=4) n-stage (n=3,6,8,10) MIN

Fig. 7 presents the case of a MIN with a large queue configuration, where the *buffer size* is b=8. The results show that the gains on *normalized throughput,* when the *buffer length* is *b*=8 are lower at all network setups (*n*=3, 6, 8, and 10), but still considerable. According to the above diagram the gains of a MIN using an internal priority vs. non priority scheme are 0.8%, 2.7%, 3.0%, and 3.5%, when *n*=3, 6, 8, and 10 respectively, under full load traffic. It is worthy of remark, that the *normalized throughput* is improved for both single and internal priority MINs due to the increment of *buffer size* (*b*=8), which is more obvious in the case of heavy traffic (λ>0.7) offered load.

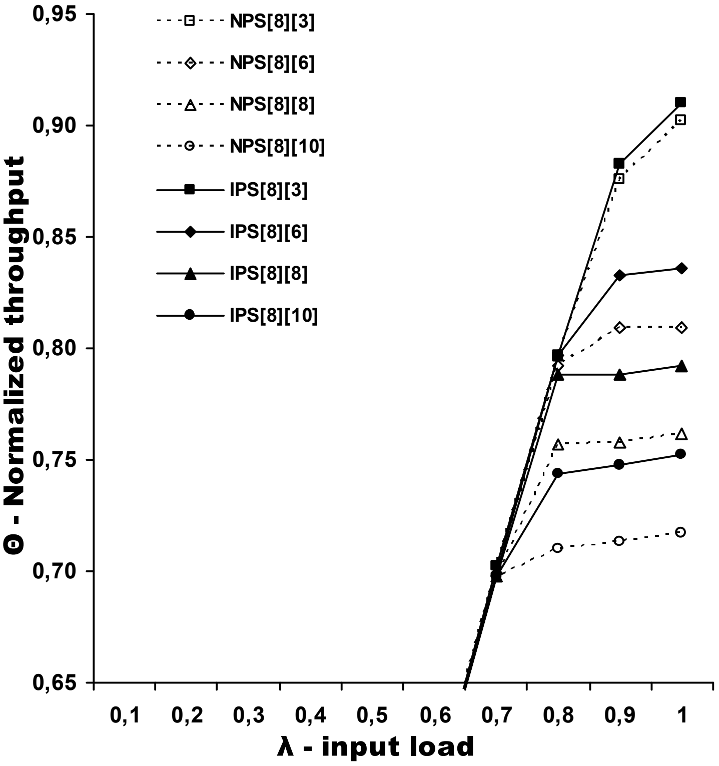

Fig. 7. Normalized throughput of a finite-buffered (*b*=8) *n*-stage (*n*=3,6,8,10) MIN

Fig. 8 represents the corresponding increments on *normalized packet delay* for internal priority vs. single priority packets of a 6-stage MIN, under different *buffer size* schemas (*b*=1, 2, 4, and 8), which are found to be negligible for all configuration setups. It emerges that when the *buffer size* of the MIN has the maximum value (*b*=8) the *normalized delay* of internal priority packets under full load traffic increases from

5.63 - the corresponding *normalized delay* of single priority packets - to 6.02, that is just the worst case. It is obvious that the corresponding single buffered (*b*=1) MINs have the same values for all performance factors at both single and internal priority schemas. The reason is that, when two packets at a stage contend for the same buffer at the next stage and there is not adequate free space to be stored the algorithm of solving the contention is the same for both single and internal priority schemas, because all queues can hold only one packet and thus, one of them is selected randomly independently of the priority scheme. It is also noteworthy that larger buffers introduce larger delays, because packets fill the buffers and stay in the network longer, thereby increasing queuing delays. Large packet delay values can adversely affect applications sensitive to packet delay or jitter, such as streaming media traffic.

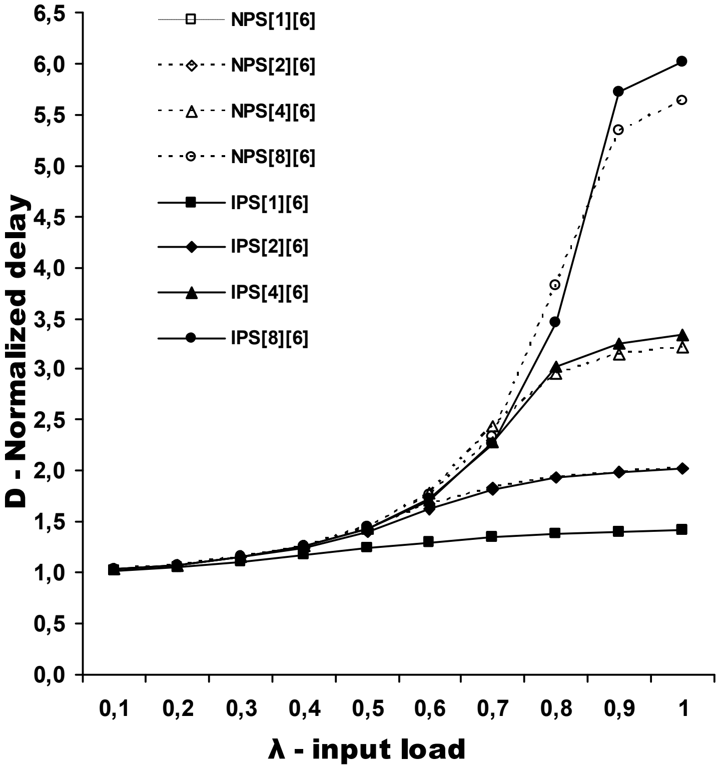

Fig. 8. Normalized delay of a finite-buffered (*b*=1,2,4,8) 6-stage MIN

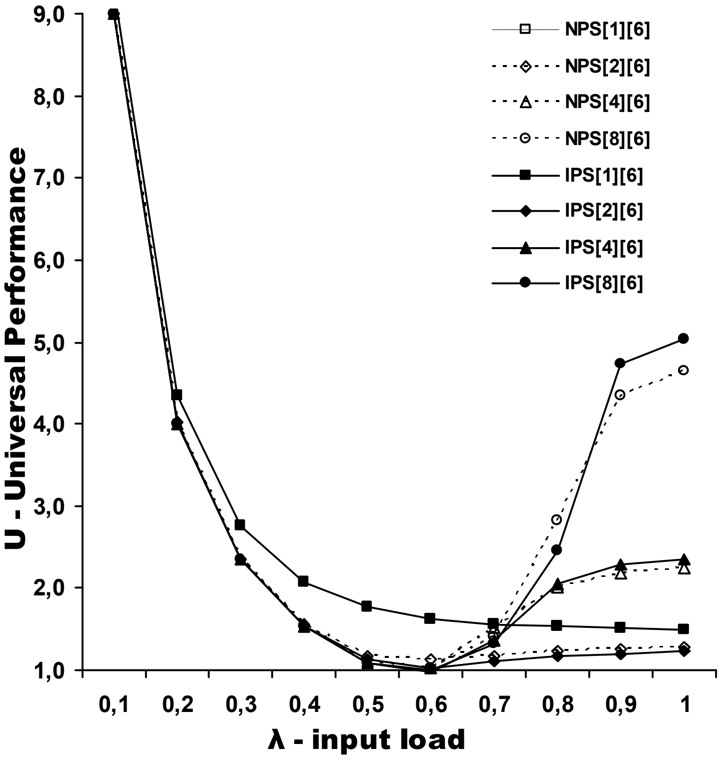

Fig. 9. Universal Performance of a finite-buffered (b=1,2,4,8) 6-stage MIN

Fig. 9 illustrates the relation of the combined *performance indicator U* of a 6-stage MIN to the *offered load λ,* under different *buffer size* configurations (b=1, 2, 4, and 8). Recall from *section 3*, the combined *performance indicator U* depicts the overall performance of a MIN, considering the weights of each individual performance factor (throughput and packet delay) are of equal importance. It is clear that the *performance indicator U* has lower (better) values as the *buffer length* increases, but when the *buffer size* reaches the values b = 4, 8 the *performance indicator U* deteriorates significantly , under moderate and heavy traffic (λ>6) using either internal or single priority scheme. This holds because delay in these cases increases rapidly, while gains in throughput are very small.

Conclusions

In this paper we have presented a novel MIN architecture employing an internal priority scheme to resolve contentions. The performance of the proposed scheme has been evaluated through simulation and compared against the performance of single-priority MINs, considering different *offered loads*, *buffer lengths* and MIN *sizes*. It has been found that the gains for MINs in terms of *throughput* using the internal priority scheme are considerable in all cases. It has to be noted that the performance gains obtained by applying the proposed techniques are in the range of 2%-5%, thus not very high in absolute numbers; taking however into account that these gains are referred against to the optimal values which are the maximum ones and achieved at virtually no cost (only some logic needs to be added to each MIN switching element to take into account the sending switching element's queue length), the proposed mechanism is definitely valid and viable. Especially, the improvement of the *throughput* is of great worth when the offered load includes mainly data packets (vs. voice packets, which are more sensitive to *packet delay*), because *throughput* is the most important performance factor in the case of data packets. It is also worth noting, that the corresponding increments of *packet delays* are negligible for all configuration setups. Moreover, the overall *performance indicator U* of a MIN, a metric combining both throughput and delay, is improved.

In this study, when calculating the value of *U*, we have considered the individual performance factors (*throughput* and *packet delay*) to be of equal importance. This is not necessarily true for all application classes, e.g. for batch data transfers throughput is more important, whereas for streaming media the delay must be optimized. The proposed "internal prioritization" can potentially be applied in other communications configurations where transmission queues are employed, thus engineers designing relevant communication infrastructures can consider to incorporate the proposed mechanism. In our future work we will consider such cases and will make efforts to provide MIN designers with metrics that will support them in choosing the best MIN setup, taking into account the applications that the MIN will support. The combination of the internal priority scheme presented in this paper with externally defined priorities will also be considered.

References

[1] Chang-hoon Choi, Sung-chun Kim. Hierarchical multistage interconnection network for shared-memory multiprocessor system. Proceedings of the 1997 ACM Symposium on Applied Computing (1997) pp. 468-472.

[2] Josep Torrellas, Zheng Zhang. The Performance of the Cedar Multistage Switching Network. IEEE Transactions on Parallel and Distributed Systems, 8(4) (1997) pp. 321-336.

[3] Gheith A. Abandah and Edward S. Davidson. Modeling the communication performance of the IBM SP2. In Proceedings of the 10th International Parallel Processing Symposium (IPPS'96); Hawaii. IEEE Computer Society Press (1996).

[4] Elizabeth Suet Hing Tse. Switch fabric architecture analysis for a scalable bi-directionally reconfigurable IP router. Journal of Systems Architecture: the EUROMICRO Journal, 50(1), (2004) pp. 35-60.

[5] Toshio Soumiya, Koji Nakamichi, Satoshi Kakuma, Takashi Hatano, and Akira Hakata. The large capacity ATM backbone switch "FETEX-150 ESP". Computer Networks, 31(6) (1999) pp. 603–615.

[6] Ra'ed Y. Awdeh and H. T. Mouftah. Survey of ATM switch architectures. Computer Networks and ISDN Systems 27 (1995) pp. 1567–1613.

[7] G. F. Goke, G.J. Lipovski. Banyan Networks for Partitioning Multiprocessor Systems. Proc. 1st Ann. Symp. on Computer Architecture (1973) pp. 21-28.

[8] G. Bolch, S. Greiner, H. de Meer, K. S. Trivedi. Queueing Networks and Markov Chains - Modeling and Performance Evaluations with Computer Science Applications. John Wiley and Sons, New York (1998).

[9] A. Merchart. A Markov chain approximation for analysis of Banyan networks. Proceedings of the ACM Sigmetrics Conference on Measurement and Modelling of Computer systems (1991).

[10] R. German. Performance Analysis of Communication Systems. John Wiley and Sons (2000).

[11] P. J. Haas. Stohastic Petri Nets. Springer Verlag (2002).

[12] C. Lindermann. Performance Modelling with Deterministic and Stohastic Petri Nets. John Wiley and Sons (1998).

[13] S.H. Hsiao and R. Y. Chen. Performance Analysis of Single-Buffered Multistage Interconnection Networks. 3rd IEEE Symposium on Parallel and Distributed Processing (1991) pp. 864-867.

[14] T.H. Theimer, E. P. Rathgeb and M.N. Huber. Performance Analysis of Buffered Banyan Networks. IEEE Transactions on Communications, vol. 39, no. 2 (1991) pp. 269-277.

[15] M.Jurczyk. Performance Comparison of Wormhole-Routing Priority Switch Architectures. Proceedings of the International Conference on Parallel and Distributed Processing Techniques and Applications 2001 (PDPTA'01); Las Vegas (2001) pp. 1834-1840.

[16] J.Turner, R. Melen. Multirate Clos Networks. IEEE Communications Magazine, 41, no. 10 (2003) pp. 38-44.

[17] M. Atiquzzaman and M.S. Akhatar. Efficient of Non-Uniform Traffic on Performance of Unbuffered Multistage Interconnection Networks. IEE Proceedings Part-E (1994).

[18] T. Lin, L. Kleinrock. Performance Analysis of Finite-Buffered Multistage Interconnection Networks with a General Traffic Pattern. Joint International Conference on Measurement and Modeling of Computer Systems. Proceedings of the 1991 ACM SIGMETRICS conference on Measurement and modeling of computer systems, San Diego, California, United States (1991) pp. 68 - 78.

[19] Cisco Systems, http://www.cisco.com/en/US/prod/collateral/routers/ps5763/prod_brochure0900aecd800f8118.pdf (2010).

[20] Cisco Systems, http://newsroom.cisco.com/dlls/2004/next_generation_networks_and_the_cisco_carrier_routing_system_overview.pdf (2004).

[21] Stevens W. R., TCP/IP Illustrated: Volume 1. The protocols, 10th Edition, Addison-Wesley Pub Company (1997).

[22] D-Link. DES-3250TG 10/100Mbps managed switch.
ftp://download.intel.com/support/express/switches/53x/530T_UG.pdf (2006).

[23] Intel Corporation. Intel Express 530T standalone switch.
ftp://download.intel.com/support/express/switches/53x/530T_UG.pdf (2010).

[24] J.S.C. Chen and R. Guerin. Performance study of an input queueing packet switch with two priority classes. IEEE Trans. Commun. 39(1) (1991) pp. 117–126.

[25] S. L. Ng and B. Dewar, Load sharing replicated buffered banyan networks with priority traffic Connecting the System: Australian Telecommunication Networks and Application Conference, Monash University, Clayton, Victoria (1995) pp. 77-82.

[26] Stevens W. R., TCP/IP Illustrated: Volume 1. The protocols, 10th Ed., Addison-Wesley Pub Company (1997).

[27] Siu-Cheung Chau, Tiehong Xiao, Ada Wai-Chee Fu. Routing and Scheduling for a Novel Optical Multistage Interconnection Network. Proceedings of the Euro-Par 2005 Parallel Processing Conference, LNCS Vol. 3648/2005 (2005) pp. 984-993.

[28] G. B. Adams and H. J. Siegel, The extra stage cube: A fault-tolerant interconnection network for supersystems. IEEE Transactions on Computers,. 31(4)5 (1982) pp. 443-454.

[29] J.H. Patel. Processor-memory interconnections for mutliprocessors. Proceedings of 6th Annual Symposium on Computer Architecture New York (1979) pp. 168-177.

[30] H. Mun and H.Y. Youn. Performance analysis of finite buffered multistage interconnection networks IEEE Transactions on Computers, (1994) pp. 153-161.

[31] D.C. Vasiliadis, G.E. Rizos, C. Vassilakis. Performance Analysis of blocking Banyan Swithces. Proceedings of the IEEE sponsored International Joint Conference on Telecommunications and Networking CISSE 06 (2006).

[32] D. Tutsch, M.Brenner. MIN Simulate. A Multistage Interconnection Network Simulator. 17th European Simulation Multiconference: Foundations for

Successful Modelling & Simulation (ESM'03); Nottingham, SCS (2003) pp. 211-216.

[33] D.Tutsch, G.Hommel. Generating Systems of Equations for Performance Evaluation of Buffered Multistage Interconnection Networks. Journal of Parallel and Distributed Computing, 62, no. 2, (2002) pp. 228-240.

[34] Y.-C.Jenq. Performance analysis of a packet switch based on single-buffered banyan network IEEE Journal Selected Areas of Communications, (1983) pp. 1014-1021.

[35] J. Garofalakis, and E. Stergiou "An analytical performance model for multistage interconnection networks with blocking", Procs. of CNSR 2008, May (2008).

[36] D.C. Vasiliadis, G.E. Rizos, C. Vassilakis, and E.Glavas. "Performance evaluation of two-priority network schema for single-buffered Delta Network", Procs. of IEEE PIMRC' 07, Sep.(2007).

[37] D.C. Vasiliadis, G.E. Rizos, C. Vassilakis, and E.Glavas. "Routing and Performance Analysis of Double-Buffered Omega Networks Supporting Multi-Class Priority Traffic", Procs. of the third International Conference on Systems and Networks Communications IEEE press, pp.56-63 (2008).

[38] A. Pombortsis, and I. Vlahavas. "Flow control in packet-switched multistage interconnection networks", Procs of the 1992 IEEE CompEuro Conference, pp. 598 – 603.

[39] A. Pombortsis, and I. Vlahavas. "A contribution to the problem of avoiding congestion in multistage networks in the presence of unbalanced traffic". Journal of Systems and Software, Vol 26, Issue 3, September 1994, Pages 273-284