Performance Evaluation of Multicast Routing over Multilayer Multistage Interconnection Networks

D. C. Vasiliadis,[a,b] G. E. Rizos,[a,b] C. Vassilakis,[a] E. Glavas[b]
[a]*Department of Computer Science and Technology, University of Peloponnese, Greece*
[b]*Technological Educational Institute of Epirus, Greece*
dvas@uop.gr, georizos@uop.gr, costas@uop.gr, eglavas@teiep.gr

Abstract

Multilayer MINs have emerged mainly due to the increased need for routing capacity in the presence of multicast and broadcast traffic, their performance prediction and evaluation however has not been studied sufficiently insofar. In this paper, we use simulation to evaluate the performance of multilayer MINs with switching elements of different buffer sizes and under different offered loads. The findings of this paper can be used by MIN designers to optimally configure their networks.

1. Introduction

Multistage Interconnection Networks (MINs) with crossbar Switching Elements (SEs) are proposed to connect a large number of processors to establish a multiprocessor system [1]. They are also used as interconnection networks in ATM switches [2, 3], gigabit Ethernet switches [4] and terabit routers [5], for implementing the switching fabric of high-capacity communication processors. Such systems require high interconnection network performance.

Significant advantages of MINs include their low cost/performance ratio and their ability to route multiple communication tasks concurrently. MINs with the Banyan [6] property e.g. Delta Networks [7], Omega Networks [8], and Generalized Cube Networks [9] are more widely adopted, since non-Banyan MINs have generally higher cost and complexity. In the industry domain, Cisco has built its CRS-1 router [10] as a multistage switching fabric. The switching fabric that provides the communications path between line cards is 3-stage, self-routed architecture.

Broadcasting and multicasting are two important functionalities of communication infrastructure, and routing strategies for MINs as well as MIN performance under broadcast and multicast traffic have been surveyed [13][14][15][16][19]. Performance analyses [16][19], in particular, have shown that MINs tend to quickly saturate under broadcast and multicast traffic. As a response to this problem, the replication of the whole MIN network or certain stages of it has been suggested, leading to *multi-layer MINs* [20]. The degree of replication L may be constant for all stages or vary across stages; in general, higher replication degrees should be employed towards the later stages of the MIN to provide the increased switching capacity needed there due to the fact that multicast and broadcast packets are "cloned" in appropriate SEs, in order to reach all intended destinations. Replication at first stages is either not employed or kept low, to minimize the MIN cost.

The performance of multilayer MINs under broadcast and multicast traffic has not however been studied insofar. In this paper, we extend the works for MIN performance prediction and evaluation (e.g. [11], [12]) to include multilayer MINs, considering both the full and the partial multicast policies [21]. We also consider different SE *buffer sizes* and *traffic loads*, offering insight to MIN designers for configuring their MIN to best meet the performance and cost requirements under the anticipated *traffic load* and quality of service specifications.

The remainder of this paper is organized as follows: in section 2 we briefly analyze a multilayer MIN for supporting multicasting routing traffic. Subsequently, in section 3 we present the configuration and operational parameters considered in this paper, whereas in section 4 we present the performance evaluation metrics that are collected. Section 5 presents the results of our performance analysis, which has been conducted through simulation experiments, while section 6 provides the concluding remarks.

2. Multilayer MIN Description

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. All types [7][8][9] of blocking multistage interconnection self-routing networks are characterized by the fact that there is exactly a unique path from each

input port to each output port, which is just the Banyan property as defined in [6]. Switching in these networks is termed as "self-routing" because when a SE accepts a packet in one of its input ports, it can decide to which of its output ports it must be forwarded, depending only on the packet's destination address.

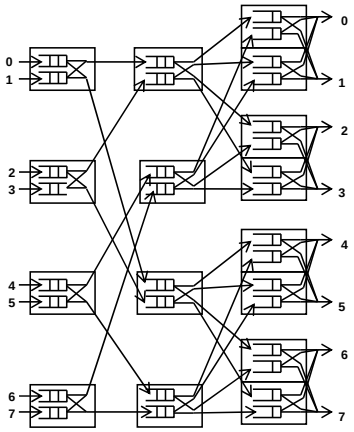

Figure 1. (8X8) Multilayer MIN, in which only the third stage is replicated

Figure 1 illustrates the modeling of an example multi-layer Delta Network. The illustrated network consists of two segments: the first one which is a single-layer segment, and the second one which is a multi-layer one (with 2 layers). It is worth noting that packet forwarding from stage 2 to stage 3 is blocking-free, since packets in stage-2 SEs do not contend for the same output link; packets at this stage can also be "cloned" (i.e. forwarded to both subsequent SEs in the context of a multicast routing activity), again without any blocking. This is always possible for cases where the degree of replication of succeeding stage i+1 (which we will denote as l_{i+1}) is equal to 2*l_i. If, for some MIN with n stages there exists some nb ($1 \le nb < n$) such that $\forall$k: $l_{k+1} = 2 * l_k$ ($nb \le k < n$), then the MIN operates in a non-blocking fashion for the last ($n - nb$) stages. Note that according to [20], blocking can occur at the MIN outputs, where SE outputs are multiplexed, if either the multiplexer or the data sink do not have enough capacity; in this paper however we will assume that both multiplexers and data sinks have adequate capacity.

In our study the MIN is assumed to operate under the following conditions:

- Routing is performed in a pipeline manner, meaning that the routing process occurs in every stage in parallel. Internal clocking results in synchronously operating switches in a slotted time model [18], and all SEs have deterministic service time.
- At each input of the network only one packet can be accepted within a time slot. All packets in input ports contain both the data to be transferred and the routing tag. The *Routing Address* (RA) and *Multicast Mask* (MM) are two equal-length fields occupying n bits each, where n is the number of stages in the MIN. Upon reception of a packet, the SE at stage k first examines the *k-th* bit of the MM; if this is set to 1, then the packet makes a multicast instead of a unicast transmission, forwarding the packet to both its output links. If the *k-th* bit of the MM is however set to zero, then the *k-th* bit of the RA is examined, and routing is performed as in the case of unicast MINs. It is obvious that, when all bits of the MM of a packet are set zero, the packet follows a unicast path, reaching one specific network output port. On the other extreme, when all its bits are set to one the packet is broadcasted to all output ports of the network. In all other cases, the packet will be forwarded to a group of output ports, which constitute the *Multicast Group* (MG).
- The offered load in all inputs of the network is uniform, all packets have the same size and the arrivals are independent of each other.
- There is a FIFO buffer in front of each SE enabling the packets of a message to be stored until they can be forwarded to the succeeding stage in the network.
- The backpressure mechanism deals with packets directed toward full buffers of the next stage, forcing them to stay in their current stage until the destination/s become/s available, so that no packets are lost inside the MIN.
- A SE operates with either partial or full multicasting. Multicasting is performed by copying the packets within the 2X2 SEs. According to the partial mechanism (PM) if any of destination buffers is not available, the packet is forwarded to the available destination and a copy remains at the present stage, in order to be later forwarded to the destination currently unavailable. When the full multicasting mechanism (FM) is employed, a packet is copied and transmitted when only both destination buffers are available.
- Conflicts between packets are solved randomly with equal probabilities.
- All packets are uniformly distributed across all the destinations. That means every output of the network has an equal probability of being one of the destinations of a packet.
- Packets are removed from their destinations immediately upon arrival, thus packets cannot be blocked at the last stage.

3. Configuration and Operational Parameters of the Evaluated MINs

In the work presented in this paper, we consider multi-layer MINs consisting of two segments, an initial

single-layer one and a subsequent multi-layer one, as the MIN depicted in Figure 1. The multi-layer segment is assumed to operate in a non-blocking manner, i.e. each stage has twice as many layers as the immediately preceding one. Since the operation of this segment is non-blocking, all SEs in it are considered to have only the buffer space needed to store and forward a single packet. On the other hand, the single-layer segment may employ different *buffer sizes*. Under these considerations, the operational parameters of the MINs evaluated in this paper are as follows:

Buffer-size b of a queue is the maximum number of packets that an input buffer of a SE can hold. In this paper we consider symmetric single- b=1 or double-b=2 buffered MINs, since double-buffered SEs have been reported [17] to provide optimal overall network performance. [17] reports that higher *buffer-size* configurations (b = 4, 8), lead to significantly increased delays and in elevated SE hardware cost; the latter in the case of multilayer MINs is of high importance, since the addition of layers has already lead to cost increase.

Offered load λ is the steady-state fixed probability of such arriving packets at each queue on inputs. In our simulation λ is assumed to be λ = 0.1, 0.2… 0.9, 1.

Network size 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=6, which is a widely used MIN size.

Multicast ratio m of an SE at stage k_{se} is the probability that a packet arriving to the particular SE has its k_{se}-th bit of its multicast mask (MM) set, effectively expressing the probability that an SE will do a multicast by forwarding the packet to both its outputs. In this paper m is considered to be fixed at all SEs and is assumed to be m = 0, 0.1, 0.5, 1. It is obvious that, when m = 0 or 1 all input traffic is either unicast or broadcast respectively. For intermediate values of m, the probability that a packet is unicast is equal to $(1\text{-}m)^n$, i.e. the joint probability that all bits in MM are equal to 0. The value m=0.1 for multicast ratio is considered, since it for a MIN size n equal to 6 evaluates to $(1\text{-}0.1)^6 = 0.9^6$ =53.14%, giving thus approximately equal probabilities for unicast or multicast transmission within the MIN.

4. Performance Evaluation Metrics for MINs

In this section we discuss the performance evaluation metrics used in this paper. We employ the typical *throughput*- and *delay*-related metrics, and we also consider the *Universal performance factor* introduced in [17], which combines *throughput* and *delay* into a single metric, allowing the designer to express the perceived importance of each individual factor through weights. Attention has been paid to the definition of *throughput* and *delay* for multi-layer MINs, since both the single-layered and multi-layered segments have to be considered.

4.1 Metrics for Single-layer MINs

In order to evaluate the performance of a multicasting, single-layer (N X N) MIN, we use the following metrics. Let T be a relatively large time period divided into u discrete time intervals ($\tau_1, \tau_2, \ldots, \tau_u$).

Average throughput Th_{avg} is the average number of packets accepted by all destinations per network cycle. Formally, Th_{avg} (or *bandwidth*) is defined as

$$Th_{avg} = \lim_{u \to \infty} \frac{\sum_{k=1}^{u} n_a(k)}{u} \qquad (1)$$

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

Normalized throughput Th is the ratio of the *average throughput* Th_{avg} to the number of network outputs N. Formally, Th can be expressed by

$$Th = \frac{Th_{avg}}{N} \qquad (2)$$

and reflects how effectively network capacity is used.

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

$$D_{avg} = \lim_{u \to \infty} \frac{\sum_{k=1}^{n_a(u)} t_d(k)}{n_a(u)} \qquad (3)$$

where $n_a(u)$ denotes the total number of packets accepted within u time intervals and $t_d(k)$ represents the total delay for the k^{th} packet. We consider $t_d(k) = t_w(k) + t_{tr}(k)$ where $t_w(k)$ denotes the total queuing delay for k^{th} packet, while waiting at each stage for the availability of a buffer at the next stage of the network. The second term $t_{tr}(k)$ denotes the total transmission delay for k^{th} packet at each stage of the network, that is just n*nc, where n=$\log_2 N$ is the number of intermediate 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 (i.e. zero queuing delay). Formally, D can be defined as

$$D = \frac{D_{avg}}{n * nc} \qquad (4)$$

Universal performance factor Upf is defined by a relation involving the two major above normalized factors, D and Th: the performance of a MIN is considered optimal when D is minimized and Th is maximized, thus the formula for computing the *universal factor* arranges so that the overall performance metric follows that rule. Formally, Upf can be expressed by

$$Upf = \sqrt{w_d * D^2 + w_{th} * \frac{1}{Th^2}} \qquad (5)$$

where w_d and w_{th} denote the corresponding *weights* for each factor participating in the *Upf,* designating thus its importance for the corporate environment. Consequently, the performance of a MIN can be expressed in a single metric that is tailored to the needs that a specific MIN setup will serve. It is obvious that, when the *packet delay* factor becomes smaller or/and *throughput* factor becomes larger the *Upf* becomes smaller, thus smaller *Upf* values indicate better overall MIN performance. 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 (5) can be replaced by:

$$Upf = \sqrt{w_d * \left(\frac{D - D^{\min}}{D^{\min}}\right)^2 + w_{th} * \left(\frac{Th^{\max} - Th}{Th}\right)^2} \qquad (6)$$

where D^{min} is the minimum value of *normalized packet delay (D)* and Th^{max} is the maximum value of *normalized throughput*. Consistently to equation (5), when the *universal performance factor Upf*, as computed by equation (6) is close to 0, the performance a MIN is considered optimal whereas, when the value of *Upf* increases, its performance deteriorates. Moreover, taking into account that the values of both *delay* and *throughput* appearing in equation (6) are normalized, $D^{\min} = Th^{\max} = 1$, thus the equation can be simplified to:

$$Upf = \sqrt{w_d * (D-1)^2 + w_{th} * \left(\frac{1 - Th}{Th}\right)^2} \qquad (7)$$

In the remaining of this paper we will consider both factors of equal importance, setting thus $w_d = w_{th} = 1$.

Average packet loss probability Pl_{avg} is the average number of packets rejected by all input ports per network cycle. Formally, Pl_{avg} is defined as

$$Pl_{avg} = \lim_{u \to \infty} \frac{\sum_{k=1}^{u} n_r(k)}{u} \qquad (8)$$

where $n_r(k)$ denotes the total number of packets that are rejected at all queues of SEs at the first stage of MIN during the k^{th} time interval.

Normalized packet loss probability Pl is the ratio of the *average packet loss probability* Pl_{avg} to the number of network input ports *N*. Formally, *Pl* can be expressed by $Pl = \frac{Pl_{avg}}{N}$ (9). Note that the *packet loss probability* in the case of unicast traffic is equal to (*λ-Th*), and this is the reason it does not appear in the *Upf* formula in [17] (this paper considers only unicast traffic). In this work, we will retain the definition of [17] for *Upf*, and we will consider *packet loss probability* as a separate metric for multicast traffic.

4. Metrics for multi-layer MINs

Recall from section 3 that multilayer (*N* X *N*) MINs considered in this paper consist of two segments, as illustrated in figure 1: the first one is a single-layer segment and the second one is a multi-layer segment operating in a non-blocking fashion. Let *l* be the number of layers at the last stage (output) of network. The number of multi-layer stages is then $n_{ml} = \log_2 l$ (since layers are doubled in consecutive stages in the multilayer segment), while the number of single-layer stages is $n_{sl} = n - \log_2 l = \log_2 N - \log_2 l$, where $n = \log_2 N$ is the total number of stages in the MIN.

Normalized throughput Th of an l-layer MIN can be consequently expressed as

$$Th = Th(n - \log_2 l) * (1+m)^{1+\log_2 l} \qquad (10)$$

where $Th(n\text{-}\log_2 l)$ is the *normalized throughput* at last stage of single-layer segment of MIN. The multiplier in equation (10) $[(1+m)^{1+\log_2 l}]$ effectively represents the *cloning factor* of a packet undergoing $1+\log_2 l$ transmissions across stages, with the probability of being duplicated in each transmission is *m*. Note that equation (10) holds under the assumption that no blockings may occur in the last $1+\log_2 l$ transmissions; the last one of single-layer and all of multi-layer segment.

Normalized delay D of an l-layer MIN can be similarly evaluated basing on the *normalized packet delay* $D(n\text{-}\log_2 l)$ of single-layer segment of MIN. Formally, *D* can be defined as

$$D = \frac{D(n - \log_2 l) * (n - \log_2 l) + \log_2 l}{n} \qquad (11)$$

The *normalized delay* of entire MIN transmission includes both single- and multi-layer segments. According to (4) the *average delay* of the single-layer segment can be expressed as $D_{avg}(n - \log_2 l) = D(n - \log_2 l) * (n - \log_2 l) * nc$. Subsequently, the *average delay* D_{avg} of entire *l*-layer MIN is simply augmented by the transmission delay of non-blocking, multi-layer segment which is $\log_2 l * nc$. Thus, the *normalized delay* just as expressed by equation (11) is computed by dividing the $D_{avg} = [D(n - \log_2 l) * (n - \log_2 l) + \log_2 l] * nc$ over the minimum packet delay, which is simply the transmission delay of all stages, i.e. $n*nc$.

Universal performance factor Upf of an l-layer MIN, can be expressed according to equation (6), and taking into account that $D^{min} = 1$, and $Th^{max} = 2*l$ by

$$Upf = \sqrt{w_d * (D-1)^2 + w_{th} * \left(\frac{2*l - Th}{Th}\right)^2} \qquad (12)$$

The maximum *normalized throughput* take place when the *multicast ratio* is m=1, and thus the *normalized throughput* at last stage of single-layer segment is also $Th(n\text{-}\log_2 l)$=1. At this case, the second term of equation (10) becomes $2^{1+\log_2 l} = 2 * l$, denoting that each queue of all layers within the non-blocking segment of the MIN forwards 2 packets at each time slot.

5. Simulation and Performance Results

The overall network performance of multicasting store and forward MINs was evaluated by developing a special-purpose simulator in C++, capable to operate under different configuration schemes. This type of modeling [12, 16] using simulation experiments was applied due to the complexity of the mathematical model. The simulator implements two different kinds of multicasting transmissions: i) full-multicast transmission, where a packet transmitted only when both queues of next stage SEs are able to accept the packet and ii) partial-multicast transmission where a packet can be serviced either fully at both directions or partially, being transmitted at one direction and remaining in the queue the transmission towards the other direction is completed. Several input parameters such as the *buffer-length,* the *number of input and output ports*, the *number of stages*, the *offered load*, the *multicast ratio,* and the *number of layers* were considered. Internally, each SE was modelled by two non-shared buffer queues, where buffer operation was based on the FCFS principle. All simulation experiments were performed at packet level, assuming fixed-length packets transmitted in equal-length time slots, where the slot was the time required to forward one (in the case of unicast) or two (in case of multicast) packet(s) from one stage to the next. In all cases packet contentions were resolved randomly.

Metrics such as packet *throughput,* packet *delay,* and *loss probability* were collected. 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 phase to allow the network to reach a steady state, by discarding the data from the first 10^3 network cycles, before initiating metrics collection.

5.1. Simulator validation

To validate our simulator, we modeled single-layer MINs using this simulator and compared the results obtained from it against the results reported in other works –selecting among them the ones considered most accurate- both under unicast and multicast traffic. In the case of unicast traffic (m=0) we found that all results obtained by this simulator (fig. 2, curve [FP]MB1_0) were in close agreement with the results reported in [12] (fig. 2), and -notably- as Theimer's model [18], which is considered to be the most accurate one. In all subsequent diagrams, curves *Z*MB*X_Y* denote the performance of a MIN whose SEs in the single-layer segment have *buffer size* equal to *X* and operating with *multicast ratio m* equal to *Y*. When *Z* is equal to *F*, the MIN in question operates under the FM policy, whereas when *Z* is equal to *P* the MIN operates under the PM policy. In the special case that *m=0*, the multicast policy is irrelevant since no multicasting occurs, thus both curves coincide and are denoted as [FP]MB*X_Y*. All curves refer to 6-stage MINs.

Moreover, for m=0.5, at the case of using partial multicasting policy on a single-layer, single-buffered, (64X64) MIN, we compared our measurements (fig.2 curve PMB1_0.5) against those obtained from Tutsch's Model reported in [16] (fig.8 solid curve), when all possible combinations of destination addresses for each packet entering the network were equally distributed, and we have found that both results are in close agreement (*normalized throughput* is about 75%).

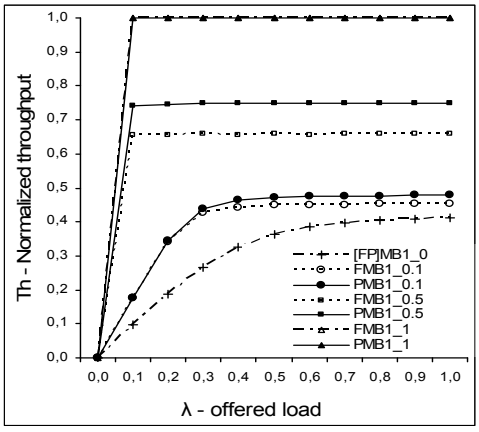

Figure 2. Normalized throughput of single-layer MINs vs. offered load

5.2. Multicasting on Single-layer MINs

In figure 2, we can observe that the partial multicasting policy offers better performance as compared to full multicasting for m=0.1 and m=0.5, while no differences are observed for m=1. We can also note that for high values of m ($m \geq 0.5$) the network is saturated (reaches its peak performance) even with very small loads ($\lambda < 0.05$), while for m=0.1 (in which case we may recall that approximately half of the packets entering the network are unicast), the network

is saturated for offered loads $\lambda \geq 0.40$.

Figure 3 illustrates the *normalized delay* in single-layer MINs. Again, the PM policy offers better performance that the FM policy for m=0.1 and m=0.5; for m=1 (i.e. only broadcast packets enter the network), the situation is reversed and the FM policy has a performance edge. This is owing to the fact that if a broadcast packet is partially served, a packet copy will remain in the queue leading thus to partially serving subsequent packets (which are broadcast packets too), and this leads to increased queuing delays. Finally, for m=1 the delay values for both multicast policies are excessively high.

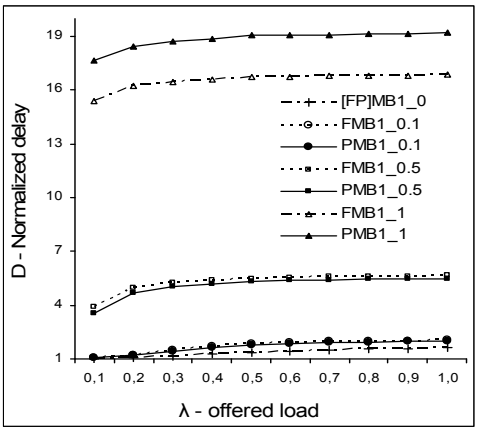

Figure 3. Normalized delay of single-layer MINs vs. offered load

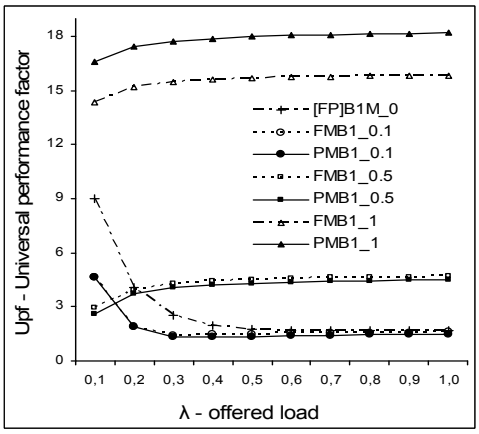

Figure 4. Upf single-layer MINs vs. offered load

Figure 4 shows the *universal performance factor* (*Upf*) for single-layer MINs. For m=1, the value of *Upf* is high (indicating poor MIN performance), and this is owing to the high delay values. For m=0 and m=0.1, the value of *Upf* drops (thus overall MIN performance increases) until the offered load reaches a value of 0.5 and 0.3 respectively. This is mainly owing to the variation of the *throughput*, which increases within the above ranges; the *delay*, on the other hand, exhibits considerably smaller variations. On the contrary, for m=0.5 and m=1 the value of *Upf* continuously increases, since the network is very quickly saturated (λ>0.1).

Figure 5 depicts the packet loss probability for single-layer MINs. We can notice that larger values of m lead to more lost packets; this is to be expected since when m increases, more packets are generated as a result of packet cloning due to multicasting, and the increased packet number cannot be successfully serviced since the network is already saturated. The PM policy has again a –marginal, in this case- edge over FM.

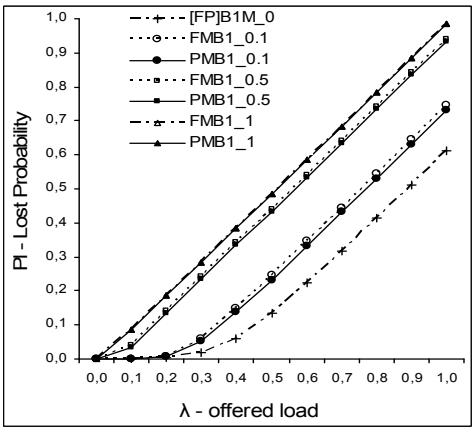

Figure 5. Loss probability of single-layer MINs vs. offered load

5.3. Multicasting on Multi-layer MINs

In this section, we present our findings for a (64x64) MIN where the number of layers at the last (6^{th}) stage l is equal to 4, i.e. the first four stages are single-layer and we multiple layers are only used at the last two stages, in an attempt to balance between MIN performance and cost. We only consider the PM policy, since it offers superior performance compared to the FM policy, as shown in the previous section. For the first 4 stages, single- and double-buffered SEs are considered, whereas at the last two stages (which are non-blocking), single-buffered SEs are used, as the absence of blockings removes the need for larger buffers.

Figure 6 shows the *normalized throughput* (*Th*) metric for the multi-layer MIN. We can easily observe a significant *throughput* increment for all values of m>0, and this is owing to the exploitation of the additional layers at the last stages, which provide the potential to route multicast packets concurrently to all their destinations. Note that *Th* increases for higher values of m, since for larger m more packets become available due to packet cloning. *Th* reaches a peak for m=1, in which

case the multiple layers in the last two stages are fully exploited. Using double-buffered queues in the single-layer segment is found to increase *throughput*, which is consistent to the findings of other works (e.g. [17]).

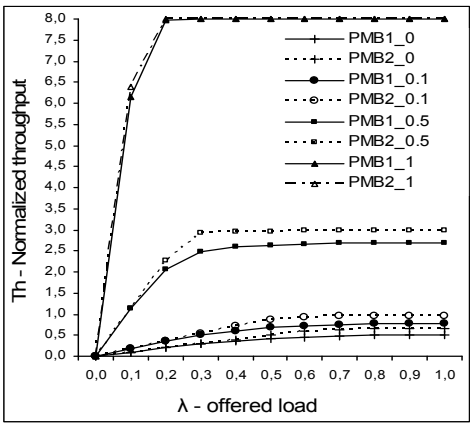

Figure 6. Normalized throughput of multi-layer MINs vs. offered load

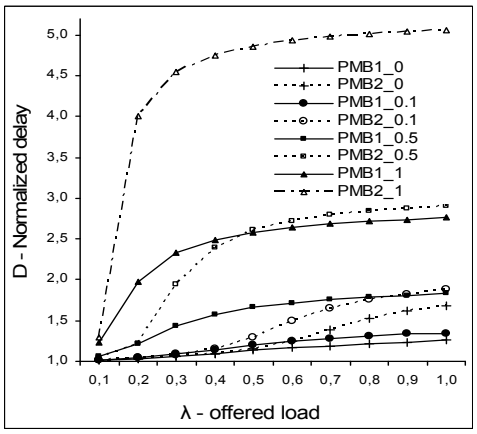

Figure 7. Normalized delay of multi-layer MINs vs. offered load

For the same reasons, *delay* drops sharply (Fig. 7) in the multi-layer MIN, especially for high values of m (50% for m=0.5 and 69% for m=1), as compared to the single-layer MIN. Using double-buffered queues in the first (single-layered) stages leads to higher delays. This increment becomes significant even at modest loads when m is high (at load $\lambda \geq 0.2$ for m=1 and at load $\lambda \geq 0.3$ for m=0.5), while for m=0.1 the *delay* increment becomes apparent at medium loads ($\lambda \geq 0.6$).

Figure 8 illustrates the *Universal performance factor* Upf in the multi-layer MIN. Note that these findings cannot be directly compared to those in fig. 4, since the maximum *normalized throughput* value in this case is Th^{max}=8, whereas in the cases shown in fig. 4 Th^{max}=1. We can use however the findings of fig. 8 to gain useful insight for the role of *buffer size* in multi-layer MINs with multicasting: for small values of m (0, 0.1), the use of double-buffered queues appears beneficial for the overall MIN performance, especially for moderate and high loads ($\lambda \geq 0.4$). For higher, however, values of m (0.5, 1), the use of double-buffered queues deteriorates overall performance, owing to the sharp increase in the *delay* factor. One extra point worth commenting is the fact that for m=0.5, in fig. 4 we can observe that *Upf* continuously increases (thus overall performance drops) with the *offered load*, whereas in figure 8 the respective curve drops in the load range $0.1 \leq \lambda \leq 0.3$, remaining almost constant for higher loads. This behavior difference can be attributed to the fact that the addition of multiple layers offers more routing capacity to the MIN, shifting thus its saturation point towards higher loads ($\lambda \approx 0.3$).

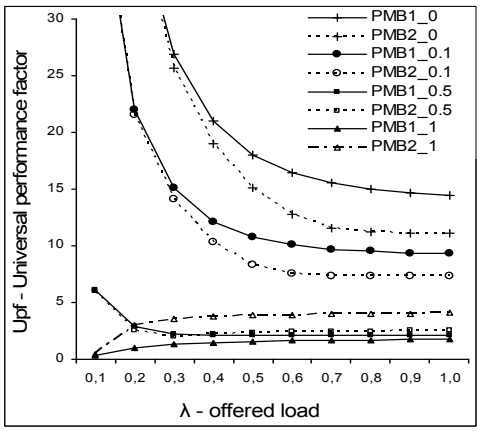

Figure 8. Universal performance factor of multi-layer MINs vs. offered load

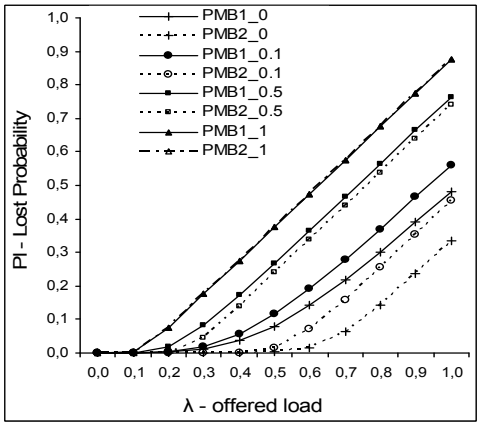

Figure 9. Loss probability of multi-layer MINs vs. offered load

Finally, fig. 9 depicts the packet *loss probability Pl* in the multi-layer MIN. It is obvious that in the multi-layer MIN these probabilities drop considerably, especially for smaller values of m, and additionally *packet loss* starts to appear at higher *offered loads* λ.

Using a double buffer is found to decrease *packet loss probability* up to 39% (m=0, λ=1), while for higher values of m this improvement is smaller, diminishing for m=1. The reduced *packet loss probability* is owing to the fact that packets entering the MIN in a double-buffered setup have a higher probability of finding a buffer position to be accommodated in, as compared to the case of a single-buffer configuration.

5.4. Multi-layer MIN Performance with Multicasting only at Multi-layer Stages

In this section we present our findings for a special operational mode of the multi-layer MIN, in which multicasting occurs only at the last $\log_2 l$+1 stages, i.e. packet cloning due to multicasting occurs only in the non-blocking segment. This mode of operation may be applied, for example, to cases of interconnected LANs, where multicasting/broadcasting can be performed within the limits of a single LAN but traffic across distinct LANs is always unicast. As an example, setting l=16 in a (64x64) MIN produces a configuration that can serve two interconnected LANs of 32 nodes each. A MIN in this mode combines both the LAN switch and the network trunk functionalities.

In the diagrams below, performance metrics are illustrated for different values of m (0, 0.1, 0.5, 1) and for l=4, thus multicasting occurs only in the last 3 stages. Since these stages are non-blocking, both *delay* and *loss probability* are not affected by the value of m and are only related to the offered load λ and *the buffer size* of the SEs in the single-layer segment. Therefore, under all values of m the *packet delay* for single- and double-buffered configurations is identical to that illustrated by curves PMB1_0 and PMB2_0, respectively, in fig. 7. Similarly, the *loss probability* for these configurations is identical to the one illustrated by curves PMB1_0 and PMB2_0 in fig. 9. For both these performance factors, we can comment that their absolute values remain low, and are even lower than the corresponding metrics collected for unicast traffic in single-layer MINs (curves [FP]MB1_0 in fig. 3 and fig. 5, respectively).

Figure 10 illustrates the *normalized throughput Th* of a MIN in which multicasting occurs only at the multilayer stages. We can observe that higher values of m lead to higher values of *Th* -similarly to the case of figure 6- since the multi-layer hardware is exploited to a fuller extent. The absolute values of *Th* in fig. 6 are higher than the ones observed in fig. 10, and this is owing to the fact that in the latter case less packets traverse the network (since packet cloning begins at stage 4, whereas at the former case packet cloning begins at stage 1). We can also observe that in the case of figure 10 the network appears to saturate for *offered load* λ=0.7 for m=1, while in the case of fig. 6 the network saturates at much lighter load (λ=0.2).

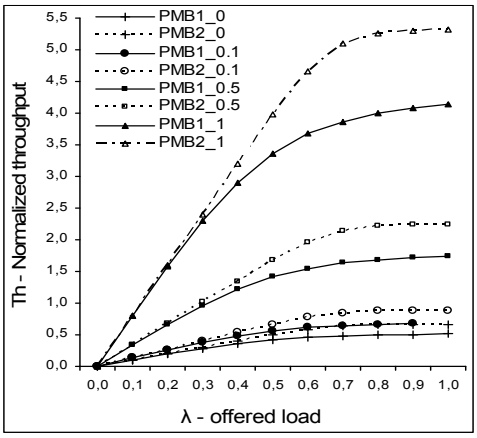

Figure 10. Normalized throughput of multi-layer MINs vs. offered load

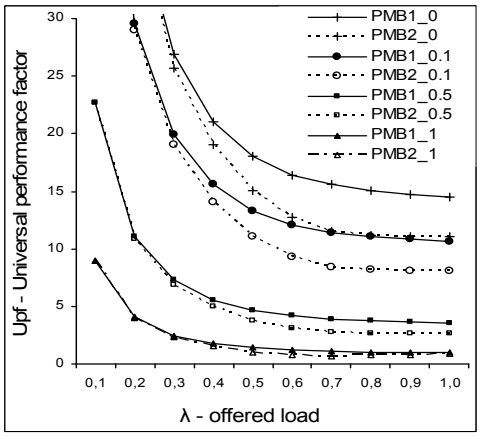

Figure 11. Universal performance factor of multi-layer MINs vs. offered load

In figure 11 the *Universal Performance Factor Upf* of a MIN in which multicasting occurs only at the multilayer stages is shown. We can observe that the overall MIN performance increases along with the offered load, and stabilizes at a moderate to high offered load (λ=0.4 for m=1; λ=0.7 for m=0 and 0.1). In all cases, using double-buffered queues in the SEs within the single layer proves beneficial for the overall performance, contrary to the case of fig. 8, where using double-buffered queues exhibit better performance that single-buffered queues only for small values of m.

6. Conclusions

In this paper we have presented an evaluation of multi-layer MINs under unicast and multicast traffic, taking into account various *offered loads*, *multicast*

probabilities, *buffer sizes* and multicasting policies. The findings of this performance evaluation can be used by network designers for drawing optimal configurations while setting up MINs, so as to best meet the performance and cost requirements under the anticipated traffic load and quality of service specifications. The presented results also facilitate performance prediction for multi-layer MINs before actual network implementation, through which deployment cost and rollout time can be minimized.

Future work will focus on examining other load configurations, including hotspot and burst loads, as well as performance evaluation under multiple priority schemes. Different multi-layer configurations, including the use of multiple layers for varying number of stages and configurations under which the number of layers within the multi-layer segment increases with a multiplication factor smaller than 2 will be also studied. In the latter case, the effect of the next layer selection algorithm [20] (random, round robin etc) on the overall MIN performance will be considered.

6. References

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

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

[3] Ra'ed Y. Awdeh and H. T. Mouftah. "Survey of ATM switch architectures". *Computer Networks and ISDN Systems* 27, pp. 1567–1613, 1995.

[4] B. Y. Yu. "Analysis of a dual-receiver node with high fault tolerance for ultra fast OTDM packet switched shuffle networks", *Technical paper, 3COM*, 1998.

[5] 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), pp. 35-60, 2004.

[6] G. F. Goke, G.J. Lipovski. "Banyan Networks for Partitioning Multiprocessor Systems" *Procs. of 1st Annual Symposium on Computer Architecture*, pp. 21-28, 1973.

[7] J.H. Patel. "Processor-memory interconnections for mutliprocessors", *Procs. of 6th Annual Symposium on Computer Architecture.* New York, pp. 168-177, 1979.

[8] D. A. Lawrie. "Access and alignment of data in an array processor", *IEEE Transactions on Computers*, C-24(12):11451155, Dec. 1975.

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

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

[11] G. Shabati, I. Cidon, and M. Sidi, "Two priority buffered multistage interconnection networks", *Journal of High Speed Networks*, pp.131–155, 2006.

[12] 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.

[13] Jaehyung Park and Hyunsoo Yoon. "Cost-effective algorithms for multicast connection in ATM switches based on self-routing multistage networks", *Computer Communications*, vol. 21, pp. 54–64, 1998.

[14] Rajeev Sivaram, Dhabaleswar K. Panda, and Craig B. Stunkel. "Efficient broadcast and multicast on multistage interconnection networks using multiport encoding", *IEEE Transaction on Parallel and Distributed Systems*, vol. 9(10), pp. 1004–1028, October 1998.

[15] Neeraj K. Sharma. "Review of recent shared memory based ATM switches" *Computer Communications*, vol. 22, pp. 297–316, 1999.

[16] D. Tutsch, G.Hommel. "Comparing Switch and Buffer Sizes of Multistage Interconnection Networks in Case of Multicast Traffic", *Procs. of the High Performance Computing Symposium*, *(HPC 2002);* San Diego, SCS, pp. 300-305, 2002.

[17] D.C. Vasiliadis, G.E. Rizos, and C. Vassilakis. "Performance Analysis of blocking Banyan Swithces", *Procs. of CISSE 06*, December, 2006.

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

[19] D. Tutsch, M. Hendler, and G. Hommel, "Multicast Performance of Multistage Interconnection Networks with Shared Buffering", *Proceedings of ICN 2001*, LNCS 2093, pp. 478–487, 2001.

[20] D. Tutsch and G. Hommel. "Multilayer Multistage Interconnection Networks", *Proceedings of 2003 Design, Analysis, and Simulation of Distributed Systems (DASD'03)*. Orlando, USA, pp. 155-162, 2003.

[21] D. Tutsch and G. Hommel. "Performance of buffered multistage interconnection networks in case of packet multicasting". *Proceedings of Advances in Parallel and Distributed Computing*, 19-21, pp. 50-57, Mar 1997.