A General Performance Model for Multistage Interconnection Networks *

C.J. Bouras, J.D. Garofalakis, P.G. Spirakis and V.D. Triantafillou

Department of Computer Engineering and Informatics,
and Computer Technology Institute,
P.O. Box 1122, 261 10 Patras, Greece
E-mail : {*bouras, garofala, spirakis, triantaf*}@cti.gr

Abstract. . In this paper we analyze the general case of Multistage Interconnection Networks (MINs), made of $k \times k$ switches with finite, infinite or zero length buffers (unbuffered). The exact solution of the steady state distribution of the first stage is derived for all cases. We use this to get an approximation for the steady state distributions in the second stage and beyond. In the case of unbuffered switches we reach the known exact solution for all the stages of the MIN. Our results are validated by extensive simulations.
Keywords: analytical models, queueing theory models, evaluation.

1 Introduction

Multistage Interconnection Networks (MINs) have attracted from the early '80s the attention of the designers of highly parallel multiprocessor systems with a large number of processors. MINs (which are packet-switched) have been adopted in the past in several machines ([2],[9]) and are expected also to play an important role in the development of high-speed networks based on Asynchronous Transfer Mode (ATM) . The performance of a MIN is of crucial importance, thus a lot of research has been dedicated to the study of how these networks perform under various conditions, through analytic techniques or simulation ([8], [10], [1], [6], [5], [7], [4]). Analytic results can be found for specific cases of MINs, which mainly rely on approximation methods.

The basic building block of the packet-switched MINs considered here, is a k-input, k-output ($k \times k$) switch grouped in stages. We examine MINs that provide a unique path from each source (processor) to each sink (memory module), which belong to the class of Banyan MINs [3]. Our work considers *general MINs*, that is, MINs made by switches with finite, infinite or zero length buffers (unbuffered), arbitrary switch size ($k \times k$) and variable injection rate p at the sources. Assuming that the traffic (requests for memory modules) is uniform,

* *This research was partially supported by the European Union ESPRIT Basic Research Projects ALCOM IT (contract no. 20244) and GEPPCOM (contract no. 9072) and the Greek Ministry of Education.*

that at each cycle a packet is generated with fixed probability p, and that packets are lost when they are attempted to be queued at a full buffer (relaxed blocking model), we derive for the general MIN, the exact steady-state distribution of queue lengths in the first stage, and of course exact formulas for the expected number of packets lost per cycle, and the mean queue length. We then use the results for the first stage and an operational approximation hypothesis to get the (approximate) distributions of the queue sizes of the second stage and beyond. Extensive simulations verify our results, as we discuss in Section 6.Our analysis, based on the theory of recurrence equations, explicitly provides the form of the queue length distribution, which is a linear mixture of geometrics.

2 Our Approach

2.1 The Model

MINs are packet switched and they are required to provide high bandwidth to support the communication between processors and memory modules. We consider that the network is built by switches connected by unidirectional lines. General MINs consist of a number of $k \times k$ switches (nodes) grouped into stages. A k-input, k-output switch, can receive packets at each of its k input ports and send them through each of its k output ports. In each output port there is a buffer. We assume that the buffers may be of infinite, finite or zero length (unbuffered switches).

If there is a unique path from each processor to each memory module then a MIN belongs to the class of Banyan Networks (BNs). We assume oblivious routing algorithms, i.e. algorithms in which the path of a packet through the network is fixed at the source node issuing it. The path can be encoded as a sequence of labels of the successive switch outputs of the path (path descriptor). Packets are generated at each processor by independent, identically distributed random processes. In our analysis we assume that each processor generates a packet with probability p at each cycle, and sends this with equal probability to any memory module (uniform access). The switches have a FIFO policy for their servers (outputs). Conflicts between packets simultaneously routed to the same output port are resolved by queueing the packet. Our analysis assumes that packets are lost when they are attempted to be queued at a full queue or in the case with unbuffered switches. In actual parallel machines, the sending processor is notified, in order to resubmit the packet later on. The service time of the output queues of each switch is assumed constant and equal to the network cycle time. The uniform access assumption allows us to represent any $k \times k$ switch as a system of k queues working in parallel, with a deterministic server each (of service time equal to 1). Any packet which enters any of the k inputs of the switch, goes with probability $1/k$ to any of the (output) queues of the switch. In our analysis we assume that the buffer length b includes the server (output). So, an unbuffered switch is referred with $b = 1$. We assume that arrivals happen at the end of each cycle (thus first the queue is served and then new packets arrive,

if any). The routing logic at each switch is assumed to be fair, i.e. conflicts are randomly resolved.

2.2 The Equilibrium and Interstage Dependencies

Most authors that have used analytic approaches for the analysis of MIN's, have remarked the basic difficulty for any analytic approach. Except for the case of unbuffered switches ([8], [6]) in all other cases, the traffic flow between consecutive stages depends upon time, that is the distribution of packet arrivals at the second and the subsequent stages is not time independent, as is the case for the first stage which is feeded by the independent "Bernoulli" processors. ([10], [6]). However, in [10] it is pointed out that the behaviour, say b_t, of a stage at time t depends mainly upon the present, a little bit $(b_{t-1}/4)$ upon the situation at time $t-1$, and is nearly independent $(b_{t-r}/4^r)$ from ancient events at time $t-r$. So, the dependency from history is exponentially desreasing. This last observation, together with the assumption that every stage of the MIN will reach an equilibrium (steady-state), leads to the marcovian approximation which we present in section 5: *The output queues of stage m that feed the stage $m+1$, are assumed to operate like independent "Bernoulli" processors with a packet generation probability equal to their utilization.* Clearly, this hypothesis equates the dynamics of the output process of a stage with its "macroscopic" averages, ignoring any time dependency of its behaviour.

3 The General Recurrence Relation for the First Stage

Let C be the random variable denoting the number of packets arriving to an arbitrary output queue of an $k \times k$ switch of the first stage, at the end of a cycle and $x_{k,c} = \mathbf{Pr}(C = c)$. Some of these arriving packets may be lost due to a full queue.

Lemma 1. *The arrival process of packets at the output queues of the first stage of the network, is given by a Bernoulli distribution $B(p/k, k)$, where p is the fixed probability of a packet generated by a processor at each cycle. Therefore we have*

$$x_{k,c} = \begin{cases} \binom{k}{c} (\frac{p}{k})^c (1-\frac{p}{k})^{k-c}, & \text{for } 0 \le c \le k \\ 0, & \text{otherwise} \end{cases} \tag{1}$$

Definition 2. Let $q^{(n)}$ be the number of packets in an arbitrary output queue at the end of the cycle n and let q be the steady state limit of $q^{(n)}$.

Definition 3. Let $v^{(n)}$ be the number of packets that are entering an arbitrary output queue at the end of cycle n and let v the steady state limit of $v^{(n)}$. It holds that $v^{(n)} \le C$ at each cycle n, when b is finite. If b is infinite, it is always true that $v^{(n)} = C$.

Definition 4. Let $p_j = \mathbf{Pr}(q = j)$, $j \geq 0$, be the distribution of q at the steady state. Also, $p_{0,1} = p_0 + p_1$

Lemma 5. *For* $0 < m \leq min(b, k)$*:*

$$\mathbf{Pr}(v^{(n)} = m) = \begin{cases} x_{k,m} & if q^{(n-1)} - \Delta(q^{(n-1)}) < b - m, \\ (x_{k,m} + x_{k,m+1} + \cdots + x_{k,k}) & if q^{(n-1)} - \Delta(q^{(n-1)}) = b - m, \\ 0 & otherwise \end{cases} \tag{2}$$

where $\Delta(q^{(n)})$ *is the departure of a packet from an arbitrary output queue at the end of cycle* n*, if any.*

Also for $m = 0$, $\mathbf{Pr}(v^{(n)} = 0) = x_{k,0}$ *for any* $q^{(n-1)}$.

Theorem 6. *The steady state flow balance equations are :* ($p_{0,1} = p_0 + p_1$)

$$\begin{aligned} p_0 &= p_{0,1}x_{k,0} \\ p_1 &= p_{0,1}x_{k,1} + p_2x_{k,0} \\ p_2 &= p_{0,1}x_{k,2} + p_2x_{k,1} + p_3x_{k,0} \\ &\vdots \\ p_k &= p_{0,1}x_{k,k} + p_2x_{k,k-1} + \cdots + p_{k+1}x_{k,0} \end{aligned} \tag{3}$$

while for $k \leq j < b$*, the general recurence holds :*

$$\begin{aligned} &p_jx_{k,0} = p_{j-k+1}(x_{k,k}) + p_{j-k+2}(x_{k,k-1} + x_{k,k}) + \cdots + \\ &p_{j-2}(x_{k,3} + x_{k,4} + \cdots + x_{k,k}) + \\ &p_{j-1}(x_{k,2} + x_{k,3} + \cdots + x_{k,k}), \quad k \leq j < b \end{aligned} \tag{4}$$

The same equation (4) holds for $j = b$*, in the case of finite buffers, or unbuffered switches* ($b = 1$)*. (Proof in full paper).*

4 Solution of the First Stage

The characteristic equation for the above recurence relation (4) is,
for $b \geq j \geq k$ ($b < \infty$ or $b = \infty$) : $F(y) = 0$, where

$$F(y) = x_{k,0}y^{k-1} - (x_{k,2} + x_{k,3} + \ldots + x_{k,k})y^{k-2} - \ldots - (x_{k,k-1} + x_{k,k})y - x_{k,k}$$

CASE 1: $F(y)$ has distinct roots $R_1, \ldots, R_{k-1}$. Then the steady-state probabilities are

$$p_j = A_1R_1^{j-1} + A_2R_2^{j-1} + \cdots + A_{k-1}R_{k-1}^{j-1} \tag{5}$$

where $A_1, A_2, \ldots, A_{k-1}$ are constants that can be derived from the initial conditions

$$\begin{aligned} p_{0,1} &= A_1 + A_2 + \cdots + A_{k-1} \\ p_2 &= A_1R_1 + A_2R_2 + \cdots + A_{k-1}R_{k-1} \\ &\vdots \\ p_{k-1} &= A_1R_1^{k-2} + A_2R_2^{k-2} + \cdots + A_{k-1}R_{k-1}^{k-2} \end{aligned} \tag{6}$$

together with $p_{0,1} = p_0 + p_1$, $\sum_{n=0}^{b} p_n = 1$, ($b < \infty$ or $b = \infty$) and the equations (3) for $p_0, p_1, \ldots, p_{k-2}$.

CASE 2: $F(y)$ has at least one multiple nonzero root. Then the system is unstable, that is $\lim_{n\to\infty} q^{(n)} = \infty$

Theorem 7 (stability criterion). *A steady state queue size distribution exists if and only if $F(y)$ has distinct roots.*

The cases of instability should occur only when $b = \infty$ (infinite buffers) and $p = 1$. However, applying our method for networks with switches 2×2, 3×3 and 4×4, we never faced the above CASE 2.

4.1 Switches with finite buffers

By applying (5) for $k = 2$, we get one root R_1, which is, given that $x_{2,0} = (1 - p/2)^2$ and $x_{2,2} = p^2/4$: $R_1 = \frac{x_{2,2}}{x_{2,0}} = (\frac{p}{2-p})^2$ The constant A_1 is given by: $A_1 = \frac{1-R_1}{1-R_1^b}$ The steady state probabilities are :

$$p_0 = A_1 x_{2,0}, p_1 = A_1(1 - x_{2,0}), p_j = A_1 R_1^{j-1}, 2 \le j \le b \text{ (for } p < 1) \quad (7)$$

or

$$p_0 = 1/4b, p_1 = 3/4b, p_j = 1/b, 2 \le j \le b \text{ (for } p = 1) \quad (8)$$

By an easy calculation, the *mean number of packets in an output queue of the first stage is*

$$E(q) = \sum_{j=0}^{b} j p_j = p + \frac{p^2[1 - p_b(1 - p + b)]}{4(1 - p)}, \text{ for } p < 1 \text{ and } b > 1 \quad (9)$$

and $E(q) = \sum_{j=0}^{b} j p_j = \frac{b+1}{2} - \frac{1}{4b}$, for $p = 1$ and $b > 1$

It is worth pointing out that for $b \to \infty$, we get $p_b \to 0$ much faster, thus equation (9) agrees with the known formula of [6] for the infinite buffer case (equation [19]).

For the *mean number of packets lost in a cycle at an output queue of the first stage* we have:

$$\underline{\text{for } p < 1} : E(\text{packets lost in one cycle}) = \begin{cases} (p^2/4)p_b, & b > 1 \\ p^2/4, & b = 1 \end{cases} \quad (10)$$

$$\underline{\text{for } p = 1} : E(\text{packets lost in one cycle}) = \begin{cases} 1/4b, & b > 1 \\ 1/4, & b = 1 \end{cases} \quad (11)$$

4.2 Switches with infinite buffers

In this case we have $b = \infty$, thus for $k = 2$, we get $x_{2,0} = (1 - p/2)^2, x_{2,2} = p^2/4$ and the root $R_1 = (\frac{p}{2-p})^2$. The difference is in the constant A_1 which is now : $A_1 = 1 - R_1$

The steady-state probabilities are :

$$p_0 = 1 - p, p_1 = A_1(1 - x_{2,0}), p_j = A_1 R_1^{j-1}, j \ge 2 \text{ for } p < 1 \quad (12)$$

For $p = 1$ we don't have steady-state probabilities, since this is an instability case. Equations (12) are in agreement with [6], since they provide the known result: $E(q) = p + p^2/4(1 - p)$

4.3 Unbuffered switches

For the general case ($k \times k$ switches), we have two balance equations :

$$\begin{aligned} p_0 &= p_0 x_{k,0} + p_1 x_{k,0} = x_{k,0} \\ p_1 &= p_0(x_{k,1} + x_{k,2} + \cdots x_{k,k}) + p_1(x_{k,1} + x_{k,2} + \cdots x_{k,k}) \\ &= 1 - x_{k,0} \end{aligned} \tag{13}$$

Since $x_{k,0} = (1 - p/k)^k$, we have

$$p_1 = 1 - (1 - p/k)^k \tag{14}$$

Equation (14) is exactly the equation $P_{m+1} = 1 - (1 - P_m/k)^k$ of [8] and [6], when $m = 0$. We may remark here, that the above authors, derive this equation for all the stages of the network. This is an evidence that our approximation for the stages beyond the first stage (section 5) is valid even for the cases when $b > 1$. Easily, we get

$$\begin{aligned} E(q) &= 1 - x_{k,0} = 1 - (1 - p/k)^k \\ E(lost) &= p - 1 + x_{k,0} = p - 1 + (1 - p/k)^k \end{aligned} \tag{15}$$

The last equation is the same with (10) for $b = 1$, when $k = 2$.

5 Subsequent Stages and Network Performance

In accordance to the remarks stated in Section 2.2, we assume now the following approximation hypothesis :

Hypothesis : The output queues of stage m that feed stage $m + 1$, are assumed to operate like processors with a packet generation probability $p_{(m)}$ such that

$$p_{(m)} = \text{utilization of an output queue of stage } m \text{ (and } p_{(0)} = p)$$

This hypothesis equates the dynamics of the output process of a stage with its "macroscopic" averages.

Definition 8. Let $p_{j,i}$ = the steady state probability of finding j packets in an output queue of stage i of the network.

Suppose that we have a network with L stages. Our approximation scheme is iterative and is described in following Algorithm I.

$p_{(0)} := p$
FOR $i = 1$ TO L DO
BEGIN
    Set $p := p_{(i-1)}$
    Calculate $x_{k,0}, x_{k,1}, \ldots, x_{k,k}$,
    Evaluate $p_{0,i}, p_{1,i}, \ldots, p_{b,i}$, from equations (5),(6)

Evaluate $E(q)$, $E(lost)$ for stage i
$p_{(i)} := 1 - p_{0,i}$
END
CALCULATE NETWORK PERFORMANCE MEASURES
(BANDWIDTH, AVERAGE TRANSIT TIME etc.)

This approximation scheme has the following nice properties :

- It provides an *exact* solution for all stages for unbuffered networks as we commented in Section 4.3
- It approximates not only the average measures such as $E(q)$ and $E(lost)$, but also the distribution itself of the queue sizes, with a maximum relative error in all cases, less than 5%. Higher errors are observed only in cases where the absolute values are very small and the simulation experiments count only a few respective events (e.g. lost packets when p is small).

6 Comparison with Simulation Results and Discussion

We performed extensive simulations to validate our results. The simulations verify our analysis for the first stage and the subsequent ones for all different cases (unbuffered, infinite and finite buffers). Moreover, they prove that the hypothesis introduced has a strong physical sence.

The comparison of the analytic results with the simulation experiments, confirms the exact solution of the first stage for all classes of MINs studied and the fact that the algorithm for the next stages presents an exact solution for all stages in the case of unbuffered network ($b = 1$). Our approximation predicts cumulative performance measures (such as mean queue length) with very small relative error. As far as the steady state distribution of queue sizes is concerned, we approximate the largest steady state probabilities with a very good accuracy, in all stages. For the low-valued probabilities (p_b, p_{b-1}) we odserve a small absolute error and a greater relative one. This error is caused probably due to the fact that the blocking phenomena that relate to these probabilities happen rarely, thus they are encountered a few times by the simulation of the network. The relative maximum relative error of 5% observed for the above probabilities, could cause a relative error of about 10% for the mean number of lost packets per queue, for the stages beyond the first since it depends mainly on those small probabilities. It is interesting to note that under our analysis networks with 2×2 switches seem to perform better than the 3×3 switches, with respect to the mean number of packets lost per queue.

- For networks with 2×2 switches :
  For low traffic ($p \leq 0.4$) buffers of size 3 are sufficient to allow only a small fraction (of about 0.0001) of the packets to be lost per queue. The buffer size becomes $b = 8$ for moderate to heave traffic ($0.4 < p \leq 0.8$) and $b = 15$ for very heavy traffic ($0.8 < p \leq 0.9$), respectively in order to keep the losses at the same low level.

- For networks with 3×3 switches :
  The buffers should be respectively of length $b = 4, b = 10, b = 18$ in order to get the same proportion of lost packets per cycle.

We expect that this tendancy - as k increases the mean number of packets lost increases - also holds for networks with greater k. The small fraction of lost packets implies that resubmission of those packets from the processors will not increase the input traffic noticaably. Thus, one can use our analysis to predict the performance of actual networks where lost packets are resubmitted later.

References

1. C. Bouras, J. Garofalakis, P. Spirakis and V. Triantafillou, *Queueing Delays in Buffered Multistage Interconnection Networks*, Proc. of the ACM Sigmetrics Conference 1987, pp. 111-121
2. A. Gottlieb, R. Grishman, C. P. Kruscal, K. P. McAuliffe, L. Rudolph, M. Snir, *The NYU Ultracomputer-Designing an MIMD Shared Memory Parallel Computer*, IEEE Trans. Computers, Vol. C-32, No. 2, Febr. 1983, pp. 175-189
3. G.F. Goke, G.J.Lipovski *Banyan Networks for Partitioning Multiprocessor Systems*, Proc. 1st Ann. Symp. on Computer Architecture, 1973, pp. 21-28
4. J. Garofalakis, P. Spirakis *The performance of Mutlistage Interconnection Networks with Finite Buffers*, Proc. of the ACM SIGMETRICS Conference, 1990, short paper.
5. R.R. Koch *Increasing the size of a Network by a constant factor Can Increase Performance by More Than a Constant Factor*, IEEE Symp. on Found. of Comp. Sc. (FOCS 88), pp. 221-231
6. C.P. Kruskal, M. Snir *The performance of multistage interconnection networks for multiprocessors*, IEEE Trans. Comp., vol. C-32, Dec 1983, pp. 1091-1098
7. C.P. Kruskal, M. Snir, A. Weiss *The Distribution of Waiting Times in Closed Multistage Interconnection Networks*, IEEE Trans. on Computers, vol. 32, 1988, p. 1337-1352
8. J.H.Patel, *Performance of processor-memory interconnection for multiprocessors* IEEE Trans. on Computing, vol. C-30, 1981, pp. 771-780
9. G. Pfister, M. C. Brantley, D. A. George, S. L. Harvey, W. J. Kleinfelder, K. P. McAuliffe, E. A. Melton, V. A. Norton, J. Weiss, *The IBM Research Parallel Processor Prototype (RP3): Introduction and Architecture*, Proc. 1985 Int. Conf. Parallel Processing, pp. 764-771
10. R. Rehrmann, B. Monien, R. Luling, R. Diemann, *On the Communication Throughput of Buffered Multistage Interconnection Networks*, ACM SPAA'96, pp. 152-161