Telecommunication Systems 10 (1998) 135–147

A virtually nonblocking self-routing permutation network which routes packets in O($\log_2 N$) time

G.A. De Biase and A. Massini

Dipartimento di Scienze dell'Informazione, Università di Roma la Sapienza, 113 Via Salaria, 00198 Roma, Italy

Asymptotically nonblocking networks are O($\log_2 N$) depth self-routing permutation devices in which blocking probability vanishes when N (the number of network inputs) increases. This behavior does not guarantee, also for very large N, that all information always and simultaneously reaches its destination (and consequently that a whole permutation passes through the device) which is a requirement of the PRAM machine. In this work the conditions for which an asymptotically nonblocking network becomes asymptotically *permutation* nonblocking are studied, finally a *virtually* nonblocking device is obtained by a retransmission procedure which guarantees that all permutations always pass through this permutation device.

1. Introduction

In massive multiprocessor systems the interchange of information among system elements (PEs and memories) is a major problem. If the model of PRAM-EREW machine is used, the interconnection device must realize N simultaneous one-to-one paths (where N is the number of multiprocessor PEs) between processors and memories. The PRAM model is synchronous and requires that these paths have O(1) depth, namely all connection requests must be satisfied simultaneously in constant time. These requirements are satisfied only by means of very expensive interconnection structures (e.g., crossbar, completely connected network) that are not suitable for large N.

In some recent works, self-routing interconnection structures with O($\log_2 N$) depth and O($N \log_2^2 N$) topological complexity have been introduced [3,4,9]. These devices, derived from banyan networks, have *high efficiency* (their blocking probability can become very small, and consequently, the quasi-totality of information can reach its destination) but their probabilistic distributed routing algorithms do not guarantee that all permutations (the whole set of input information) always pass through the network. Among high efficiency networks, asymptotically nonblocking networks are interesting because in these devices the blocking probability vanishes when N increases.

In this work, the conditions for which asymptotically nonblocking networks, defined and studied in detail in [4], become asymptotically *permutation* nonblocking are studied and, to respect the PRAM requirement that all information always and

simultaneously reaches its destination, a retransmission procedure, which guarantees that all permutations always pass through a high efficiency network, is studied.

2. Hit and miss permutations

Let B_N be a permutation network of size N (with N inputs and N outputs) and let $I = \{i_n\}$ and $O = \{o_n\}$, $n = 1, \ldots, N$, be the sets of its inputs and outputs, respectively. B_N realizes N simultaneous one-to-one connections between each i_n and each o_n. The one-to-one mappings of I onto O are characterized by the set of all permutations $P = \{p_j\}$, $j = 1, \ldots, N!$, of the elements of I onto O.

In nonblocking permutation networks all connection requests presented at the inputs i_n reach their destinations, while in blocking networks a certain number of requests cannot be honored. In blocking networks the ratio $pb_N = (r_{\text{in}} - r_{\text{out}})/r_{\text{in}}$, where r_{in} is the number of simultaneous connection requests (input) and r_{out} is the number of nonblocked requests (outputs), is the blocking probability of a B_N [3,7–9] (pb_N has the subscript N because it can depend on N) and represents the probability that a request at the generical input i_n cannot reach its destination o_n when N requests are simultaneously applied on the whole input set I. The quantity

$$\eta_N = 1 - pb_N = \frac{r_{\text{out}}}{r_{\text{in}}}$$

is the ratio between nonblocked and entering information and it is the probability that a request at an input i_n of the B_N reaches its destination o_n when N requests are simultaneously applied on the whole input set. η_N is a measure of the nonblocking capability of a network and it will be called *efficiency* (see [4]).

2.1. Asymptotically permutation nonblocking networks

If a probabilistic routing algorithm [3,4,8,9] acts on a B_N, each request is independently routed on a path with blocking probability pb_N. In this case (see [4,5]) the probability that N requests, simultaneously presented at the input set I of a B_N, *all* reach their destinations is

$$H_N = (\eta_N)^N. \tag{1}$$

H_N represents the probability that the whole permutation p_j is realized and it is the ratio between the number of *hit* (nonblocked) permutations, P_{hit}, and the number of entering permutations, P_{in}:

$$H_N = \frac{P_{\text{hit}}}{P_{\text{in}}}.$$

H_N is the *permutation efficiency* of a B_N. Nonblocking permutation networks have $\eta_N = 1$ and, consequently, $H_N = 1$ for any N. Blocked permutations will be called *miss* permutations, and the quantity

$$Pb_N = 1 - H_N = \frac{P_{\text{miss}}}{P_{\text{in}}}$$

represents the probability that a permutation is blocked.

In [4] asymptotically nonblocking permutation networks are introduced:

Definition 1. A blocking B_N with efficiency η_N is called asymptotically nonblocking if

$$\lim_{N\to\infty} \eta_N = 1.$$

Using (1), a further definition can be introduced:

Definition 2. A blocking B_N with permutation efficiency H_N is called asymptotically permutation nonblocking if

$$\lim_{N\to\infty} H_N = 1. \tag{2}$$

3. Behavior of a set of blocking networks

Let $S_N = \{B_{N_k}\},\ k = 1, \ldots, K$, be a set of K identical and independent permutation networks B_{N_k} of size N and blocking probability pb_N, the inputs and the outputs of S_N belong to the set $I^* = \{I_k\} = \{i_{n,k}\}$ and $O^* = \{O_k\} = \{o_{n,k}\},\ n = 1, \ldots, N;\ k = 1, \ldots, K$, respectively. When $N \times K$ requests are simultaneously applied at the whole input set I^* of S_N, K permutations act simultaneously on the set S_N, and K one-to-one mappings $I \rightleftharpoons O$ are simultaneously performed. Following [4], if B_{N_k} are blocking networks, the *overall* blocking probability pb_N^* of the whole set S_N can be defined. pb_N^* is the probability that, if K requests are simultaneously presented (each one at an input i_n of each B_{N_k} network), no connection request reaches its destination. If connection requests at the inputs of each B_{N_k} are completely independent (K uncorrelated permutations simultaneously act on each B_{N_k}), pb_N^* is given by [4,5]

$$pb_N^* = (pb_N)^K. \tag{3}$$

Thus, the quantity

$$\eta_N^* = 1 - pb_N^* \tag{4}$$

is the *overall efficiency* of the set S_N and represents the probability that, if K requests are simultaneously presented each one at an input i_n of each B_{N_k} network, at least one connection request reaches its destination. In [4] it is proved that a device built

by a set S_N of blocking networks can become asymptotically nonblocking if suitable conditions on the number of networks K_N are verified (see [4, theorem 1]) and if the corresponding outputs o_n of all B_{N_k} are ORed (in this case η_N^* is the efficiency of the device [4]).

Following the same outline, the conditions for which a set S_N of blocking networks becomes asymptotically *permutation* nonblocking are studied.

Theorem 1. Let S_N be a set of identical and independent blocking networks B_{N_k}, each with blocking probability pb_N, and let K_N, depending on N, be the number of B_{N_k} networks of the set S_N. The set S_N is asymptotically permutation nonblocking if all permutations presented at B_{N_k} networks are uncorrelated, and if

$$K_N = \frac{\ln(1 - \sqrt[N]{f(N)})}{\ln pb_N}, \tag{5}$$

where $0 < pb_N < 1$ for any N and $0 < f(N) < 1$ is any function for which

$$\lim_{N\to\infty} f(N) = 1. \tag{6}$$

Proof. A device is asymptotically permutation nonblocking if condition (2) is verified, using (1) and (4) (in the case of a set S_N), condition (2) becomes $\lim_{N\to\infty}(\eta_N^*)^N = 1$. By the substitution

$$\eta_N^* = \sqrt[N]{f(N)}, \tag{7}$$

where $0 < f(N) < 1$, equation (2) becomes $\lim_{N\to\infty}(\sqrt[N]{f(N)})^N = 1$, which is evidently true when $\lim_{N\to\infty} f(N) = 1$. There follows from (4), (3) and (7) that

$$\sqrt[N]{f(N)} = 1 - (pb_N)^{K_N},$$

from which

$$K_N = \frac{\ln(1 - \sqrt[N]{f(N)})}{\ln pb_N}. \qquad \square$$

4. An asymptotically permutation nonblocking device

In [4], a $O(\log_2 N)$ depth self-routing asymptotically nonblocking device, based on stacks of K_N banyan networks, has been introduced. This permutation device is sketched in figure 1 and it consists of two parts: the first part (the randomizer) is devoted to transforming the input permutation p_j into a set of K_N uncorrelated permutations, while the second part (the router) addresses connection requests towards their destinations. The output ports o_n of the whole device are obtained by the logical OR of the corresponding output ports of all B_{N_k} (see figure 1). To easily obtain self-routing capability, this permutation device is built by three cascaded stacks, the planes of which are butterfly networks (see figure 2).

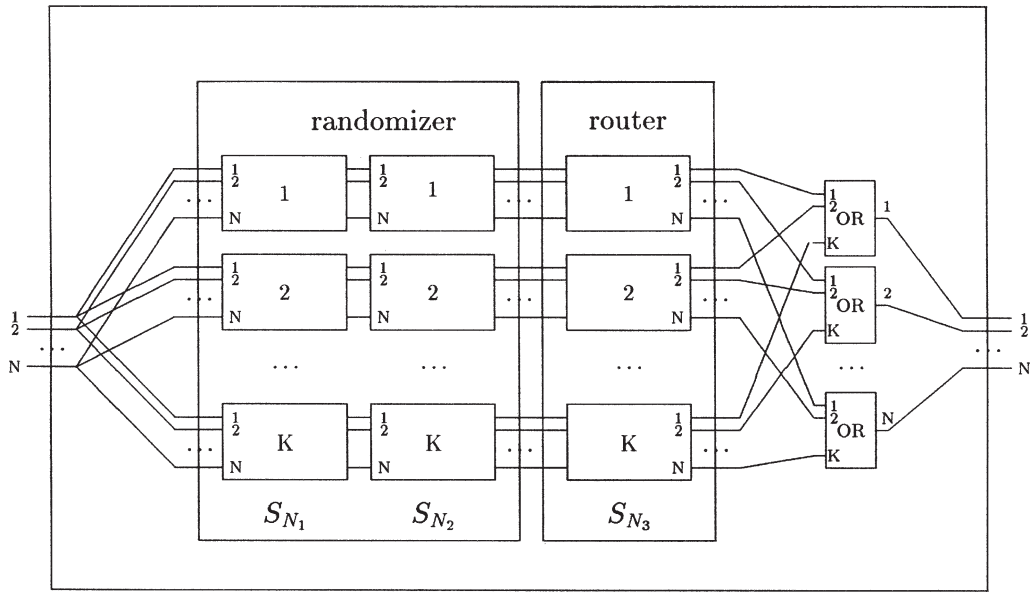

Figure 1. The permutation device. S_{N_1}, S_{N_2} and S_{N_3} are stacks of K butterfly networks.

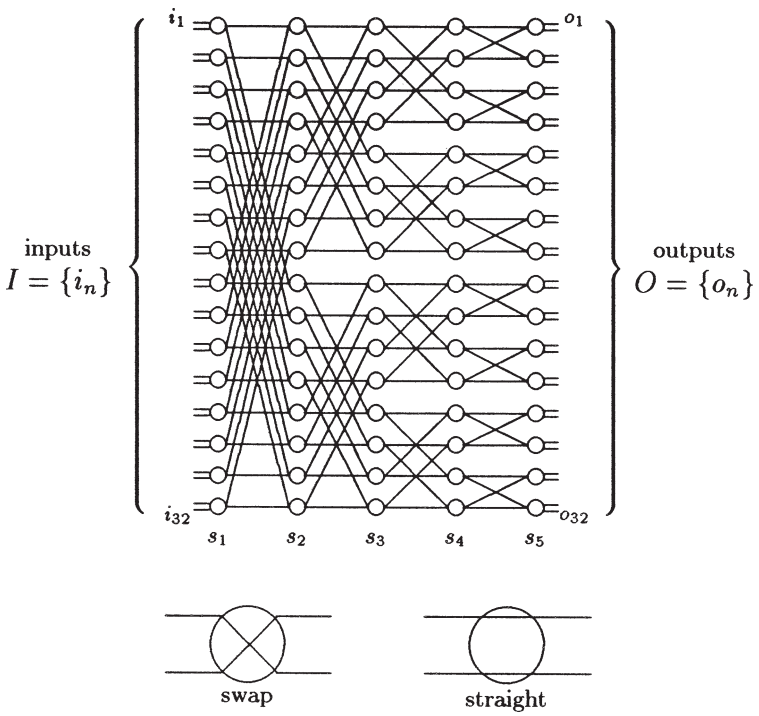

Figure 2. Butterfly network with $N = 32$ and the functions of its nodes.

To generate the set of K_N uncorrelated permutations, a way similar to that discussed in [2,9] is used. In these works it is pointed out that banyan networks, with nodes randomly set, are effective in generating random permutations. Hence, two cascaded stacks (S_{N_1} and S_{N_2} in figure 1) act as a randomizer. At the inputs of each plane of the first stack, K_N copies of the same permutation p_j are presented simultaneously. In each plane of each stack (constructed by butterfly networks) the nodes are set, at each time T, on a randomly chosen status (swap or straight). Then, on each B_{N_k}, N one-to-one connections between any input i_n and any output o_n are always obtained.

Requests are routed to their destinations by a third stack S_{N_3} on which runs the simple distributed algorithm presented in [8] which works in parallel on all planes and on all nodes stage-by-stage, namely:

- each node of a stage is set in a way that the request is routed to the upper or lower node terminal according to its binary destination address (0,1),
- if on a node two requests claim simultaneously the same terminal, the state of the node is randomly chosen, and only one request continues along its path.

The multistage structure of this permutation device and the distributed self-routing algorithm guarantee that the information wavefront synchronously passes through the network stages in $3\log_2 N - 2$ steps (see [4]). Then the system can work in pipeline

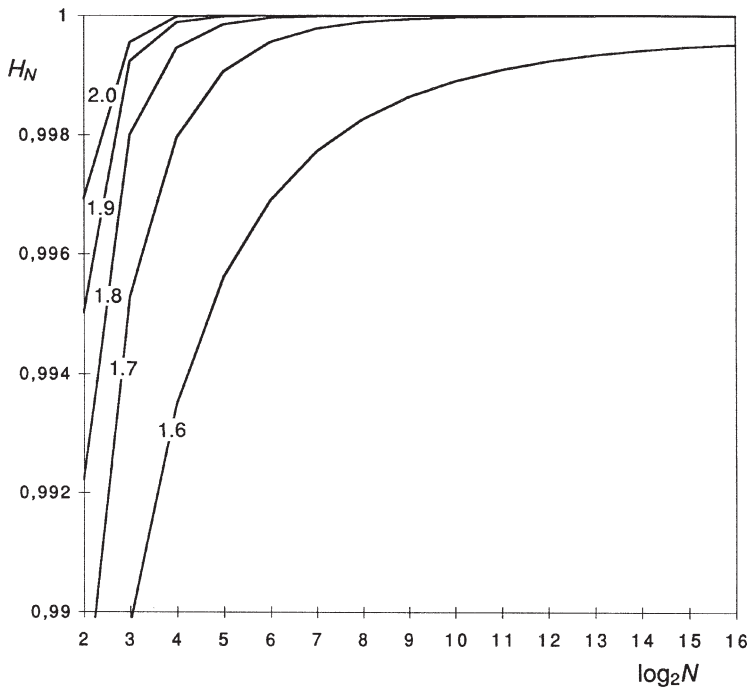

Figure 3. Permutation efficiency H_N of the presented device versus $\log_2 N$, using the function $K_N = \log_2^{\gamma} N$ ($\gamma = 1.6, 1.7, 1.8, 1.9, 2.0$).

and information can be presented at the device inputs at each time interval ΔT, where ΔT is the stage-to-stage propagation time.

4.1. Permutation efficiency

To obtain an asymptotically *permutation* nonblocking device, the condition stated in equation (5) (which gives the number of planes K_N of the routing stack and, consequently, of the randomizer) must be verified. When in an interval (N_a, N_b) the values of K_N are given by a suitable function, theorem 1 guarantees that the permutation efficiency H_N of the set S_N increases with N for all values of N belonging to the same interval. The values of blocking probability pb_N and of efficiency η_N of a banyan multistage network, under permutation requests, are given with good accuracy by the model of Szymansky and Hamacher [8]. In figure 3 the behavior of H_N, computed by the function $K_N = \log_2^\gamma N$, is shown. This function, for suitable γ values ($\gamma = 1.6, 1.7, 1.8, 1.9, 2.0, \ldots$), generates K_N values which verify the conditions stated by theorem 1 in a large interval of N. As one can see, the permutation efficiency of the device quickly increases with N.

5. Virtually nonblocking permutation networks

Theorem 1 states that a set of blocking networks S_N is asymptotically permutation nonblocking if K_N increases according to (5), and that a desired value of H_N can always be obtained for any N with a suitable choice of $f(N)$. Unfortunately it does not guarantee that *all* permutations *always* pass through S_N.

To guarantee that all permutations pass through the set S_N, miss permutations can be detected and then retransmitted. These operations generate time losses. In nonblocking networks it has not time loss because the information flux through the network is constant, and, at any time interval ΔT, a permutation appears at the network outputs (ΔT is the device crossing time or, when the system works in pipeline, the stage-to-stage propagation time). The retransmission operation modifies the time behavior of the device because for each operation a retransmission time $\mathcal{R}$ (the cost of each retransmission) is needed.

5.1. Time efficiency

Retransmitted permutations have Pb_N blocking probability too, then each miss permutation has the following m-retransmission probability Pb_{N_m}:

$$Pb_{N_m} = (Pb_N)^m, \tag{8}$$

where m ($m = 1, \ldots, \infty$) is the number of retransmissions. Thus, the probability that a permutation is retransmitted is

$$Pb_{N_{\text{retr}}} = \sum_{m=1}^{\infty} (Pb_N)^m = \frac{Pb_N}{1 - Pb_N} \tag{9}$$

and the total number of retransmitted permutations P_{retr} is

$$P_{\text{retr}} = P_{\text{in}} Pb_{N_{\text{retr}}}. \tag{10}$$

Because for each retransmitted permutation a cost $\mathcal{R}$ is spent, the *time efficiency* $\mathcal{H}_N$ will be introduced:

$$\mathcal{H}_N = \frac{P_{\text{in}}}{P_{\text{in}} + \mathcal{R} P_{\text{retr}}}. \tag{11}$$

Using (9) and (10), $\mathcal{H}_N$ becomes

$$\mathcal{H}_N = \frac{1}{1 + \mathcal{R} Pb_N / (1 - Pb_N)}. \tag{12}$$

Using retransmission, a blocking B_N with permutation efficiency H_N presents the same operating behavior of a nonblocking network (all permutations always pass through the network), but it presents a time efficiency $\mathcal{H}_N$ instead of 1.

Definition 3. A blocking B_N with time efficiency $\mathcal{H}_N$ is called virtually nonblocking if

$$\lim_{N \to \infty} \mathcal{H}_N = 1.$$

Using a suitable retransmission procedure, asymptotically permutation nonblocking networks (see definition 2) become virtually nonblocking.

5.2. Detection of miss permutations

To detect miss permutations the procedure sketched in figure 4 can be used. As one can see, copies of all connection requests arrived at the network outputs are sent back to their source by means of a second permutation network and then are compared with a stored map of the previous requests. This second network is identical to the routing one, but works in the opposite direction. When the comparison detects a request loss, the system stops and a retransmission of the whole set of requests (the whole permutation) occurs. To guarantee that these operations can be performed, during the forward phase all states of all nodes of the forward network are stored at each time T. These states are restored in the back phase on the correct nodes and at the correct time on the back network to make the return paths. For these reasons, the detection of miss permutations can be obtained after two times of the network crossing time. Obviously, if a miss permutation is detected, all operations performed during this time interval are lost.

In figure 5 the time efficiency behavior of the permutation device presented in section 4.1 is shown. The time efficiency $\mathcal{H}_N$ is computed on the router stack S_{N_3} by (11) because S_{N_1} and S_{N_2} (stacks of the randomizer) have always $\eta_N = 1$ (and consequently $H_N = 1$ and $\mathcal{H}_N = 1$), while $\mathcal{R}$ is two times the depth of the whole device. When the device works in pipeline $\mathcal{R} = 6 \log_2 N - 4$. The blocking probability

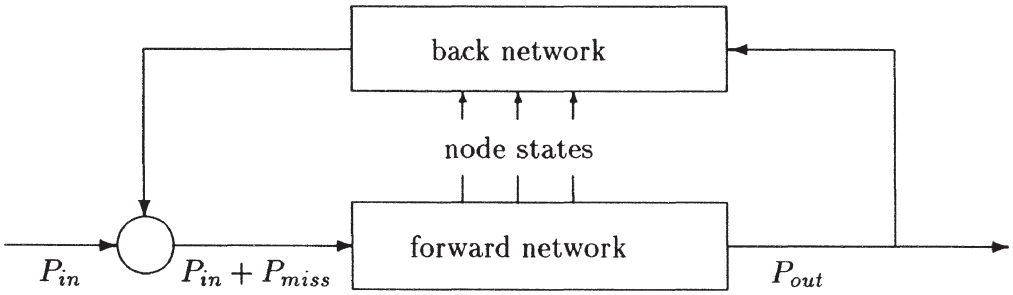

Figure 4. Retransmission of miss permutations on a probabilistic permutation network. After a delay $\mathcal{R}$, the node detects miss permutations. At the correct time, the stored node states are transmitted from the forward network to the back network to perform return paths.

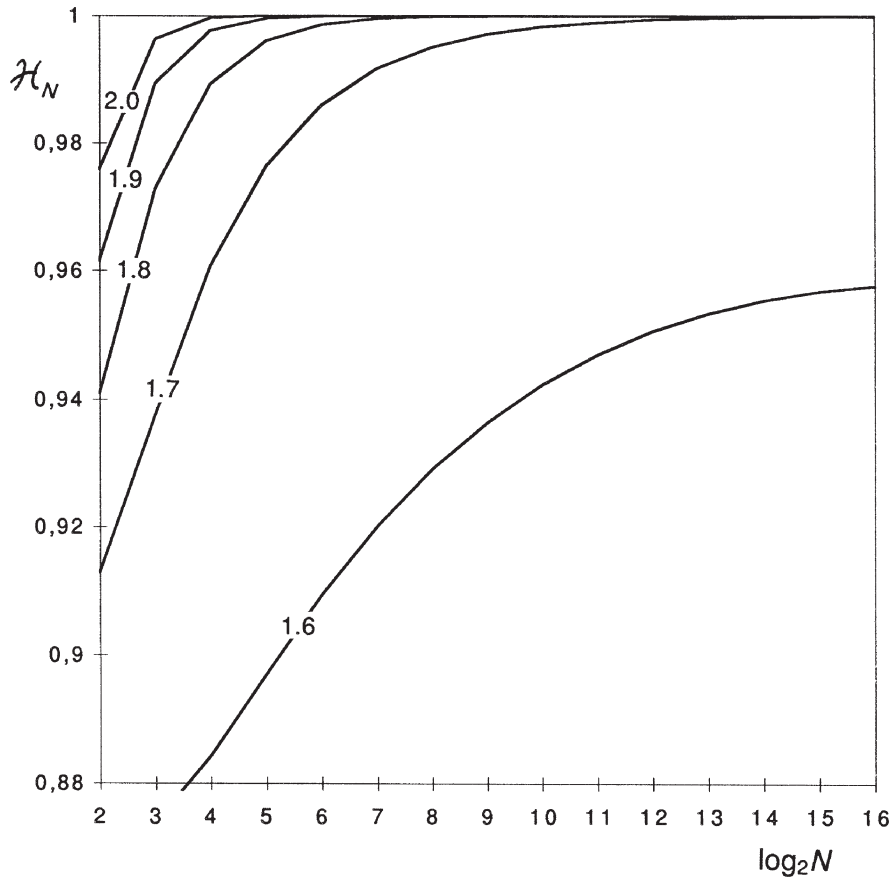

Figure 5. Time efficiency $\mathcal{H}_N$ of the presented device versus $\log_2 N$, using the function $K_N = \log_2^\gamma N$ ($\gamma = 1.6, 1.7, 1.8, 1.9, 2.0$).

of the component banyan networks is computed by the cited model of Szymansky and Hamacher [7].

6. Simulations

To verify the time efficiency behavior of the described virtually nonblocking permutation device, it has been examined by numerical simulations. Simulations give the values of the device time efficiency $\mathcal{H}_N^S$ versus $\log_2 N$ when the function $K_N = \lceil \log_2^{1.7} N \rceil$, which verifies the conditions stated by theorem 1 in the considered interval of N, is chosen (the ceiling is necessary to obtain integer K_N values). The behavior of the three stacks has been simulated by a numerical program which utilizes (for each N) as input of the whole device a number of randomly chosen permutation p_j. The randomization of requests is obtained by setting all the nodes of each plane of the stacks S_{N_1} and S_{N_2} on randomly chosen states. The simple distributed algorithm presented in section 4.1 routes requests on the stack S_{N_3}. When a request does not reach the device output the whole permutation is retransmitted. Because the rapid increase with N of

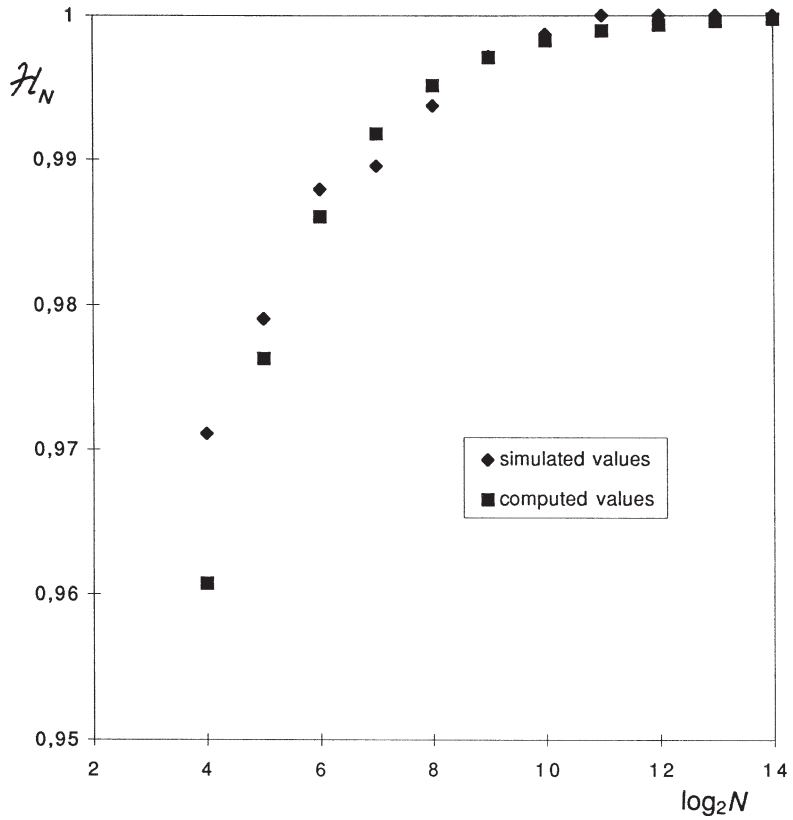

Figure 6. Comparison between computed $\mathcal{H}_N$ and simulated $\mathcal{H}_N^S$ time efficiencies for $K_N = \lceil \log_2^{1.7} N \rceil$. The values of $\mathcal{H}_N$ and $\mathcal{H}_N^S$ are scattered because K_N values are rounded up.

the number of permutations ($N!$) generates very large computation times, to obtain $\mathcal{H}_N^S$ values a number of attempts, sufficient to reach at least the 99% confidence level, has been executed in the interval $2^4 \leqslant N \leqslant 2^{13}$. Simulated values of the device efficiency $\mathcal{H}_N^S$ compared with $\mathcal{H}_N$ computed values, when $K_N = \lceil \log_2^{1.7} N \rceil$, are presented in figure 6. In this figure simulated efficiency values $\mathcal{H}_N^S$ and computed $\mathcal{H}_N$ values of the permutation device are slightly scattered because K_N values are rounded up.

7. Conclusions

Theorem 1, presented in section 3, states that a set of blocking networks S_N is asymptotically permutation nonblocking if its number of planes K_N increases according to (5) and that a desired value of the permutation efficiency H_N can always be obtained for any N with a suitable choice of K_N. The asymptotically nonblocking device, presented in section 4, with a moderate increase of its topological complexity (with respect to the requirements stated in [4, theorem 1]) can become asymptotically permutation nonblocking. When the device planes are banyan networks, this behavior is possible starting from $K_N = \log_2^{1.6} N$; in this case the topological complexity of the device is $O(N \log_2^{2.6} N)$, which is greater than that of, e.g., Koppelman–Oruç and Batcher networks [6,1], which are nonblocking but have a worse depth ($O(\log_2^2 N)$ instead of $O(\log_2 N)$).

To overcome the fact that a very small number of permutations (decreasing with N) cannot pass through the set S_N, miss permutations are detected and then retransmitted. Using the retransmission procedure all permutations always pass through the device, but the detection and the retransmission of miss permutations generate time losses which reduce the time efficiency of the system. Miss permutations can have multiple retransmissions, but, for the presented device, this situation is a very rare occurrence, in fact, in the worst case ($K_N = \log_2^{1.6} N$), sensible multiple retransmissions of miss permutations can occur every some ten years when a stage to stage propagation time $\Delta T = 10^{-9}$ sec is used (see figure 7).

The behavior of the time efficiency of this virtually nonblocking device has been examined by numerical simulations. Simulated values are strongly consistent with the computed ones and this fact confirms the validity of the assumption made in section 2.1. With a little increase in topological complexity, the desired value of permutation efficiency can be quickly reached for any N (see figure 3). This fact permits that the time efficiency can also be increased (see figure 5) reducing the mean number of multiple retransmissions (see (8)) which become very rare occurrences. Thus, sensible displacements from the behavior of nonblocking networks can occur very rarely.

In the presented virtually nonblocking device the two most important features of banyan networks are also maintained: moderate depth ($3 \log_2 N - 2$ stages) and simple request routing (obtainable by a self-routing distributed algorithm which permits pipelined operations). This device is inherently fault tolerant, in fact it consists of three

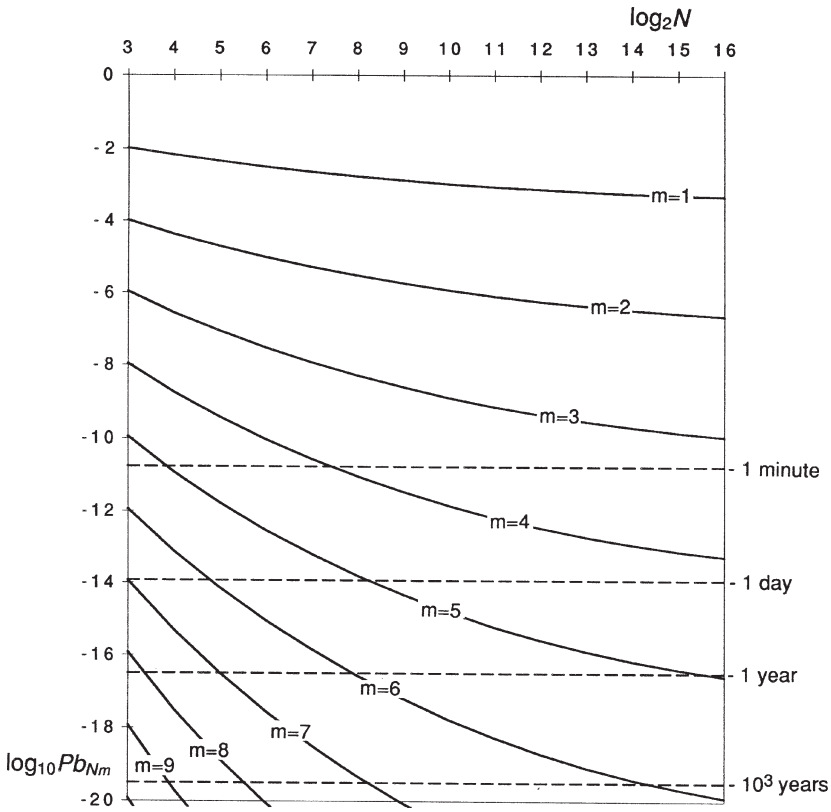

Figure 7. Occurrence of multiple retransmissions of miss permutations of the device ($\log_{10} Pb_{N_m}$ versus $\log_2 N$). m is the number of retransmissions. Values are computed in the worst case ($K_N = \log_2^{1.6} N$). The stage-to-stage propagation time is $\Delta T = 10^{-9}$ sec.

vertical stacks, each of K_N banyan networks, which implement many physical paths for each logical path. Faults on device nodes slightly modify the device efficiency as shown in [4].

Using retransmission, a blocking B_N with permutation efficiency H_N presents the same operating behavior of a nonblocking network (all permutations always pass through the network), but it presents a time efficiency $\mathcal{H}_N$ instead of 1. In the presented permutation device, the obtained time efficiency values, closer and closer to 1 for large N, guarantee that the device behavior is very close to that of nonblocking networks and, for this reason, it can successfully be used to build massive PRAM-like multiprocessor systems.

Acknowledgements

The authors thank an anonymous referee of the previous work [4] for focusing their own attention on permutation efficiency of blocking networks.

References

[1] K.E. Batcher, Sorting networks and their application, in: *Proc. of Spring Joint Computer Conference* (1968) pp. 307–314.
[2] R.L. Cruz, The statistical data fork: A class of broad-band multichannel switches, IEEE Transactions on Communications 40 (1992) 1625–1634.
[3] G.A. De Biase, C. Ferrone and A. Massini, A quasi-nonblocking self-routing network which routes packets in $\log_2 N$ time, in: *Proc. of the IEEE INFOCOM '93* (1993) pp. 1375–1381.
[4] G.A. De Biase, C. Ferrone and A. Massini, An O($\log_2 N$) depth asymptotically nonblocking self-routing permutation network, IEEE Transactions on Computers 44 (1995) 1047–1050.
[5] D.V. Huntsberger, *Statistical Inference* (Allyn and Bacon Inc., Boston, 1967).
[6] D.M. Koppelman and A.Y. Oruç, A self-routing permutation network, Journal of Parallel and Distributed Computing 10 (1990) 140–151.
[7] T.H. Szymansky and V.C. Hamacher, On the permutation capability of multistage interconnection networks, IEEE Transactions on Computers 36 (1987) 810–822.
[8] T.H. Szymansky and V.C. Hamacher, On the universality of multipath multistage interconnection networks, Journal of Parallel and Distributed Computing 7 (1989) 541–569.
[9] T.H. Szymansky and C. Fang, Randomized routing of virtual connections in essentially nonblocking $\log N$ depth networks, IEEE Transactions on Communications 43 (1995) 2521–2531.