GSE: a full-access multistage interconnection network of arbitrary size

RAJIB K. DAS† and NABANITA DAS†

Existing multistage interconnection networks (MINs) which are based on 2 × 2 switching elements are generally of size N × N where N is a power of 2. So in situations where we need to connect N processors and N resources and N is an arbitrary number (not necessarily a power of 2) we have to go for a network of size $N' \times N'$ where $N' = 2^{\lceil \log_2 N \rceil}$. This causes a wastage of resources. In order to overcome this problem we propose a new MIN called the Generalized Shuffle Exchange (*GSE*) *of size N × N where N need not be a power of 2, but only an even number. It uses N/2 switches per stage and the number of stages is equal to* $\lceil \log N \rceil$. *We show that GSE is a full-access network, i.e. every input can reach every output of the network. Routeing between any input–output pair in GSE is simple and can be done by using a routeing vector, generated from the input and output addresses. When N is a power of 2, say* 2^n, *GSE reduces to a conventional n-stage network with a unique path for each input–output pair. But, if* $2^{n-1} < N < 2^n$, *given a specific input, there are* $2^{\lceil \log N \rceil} - N$ *outputs for which there exist alternative paths. Therefore, to realize any N × N permutation in GSE, we are to select a set of N conflict free paths, one for each input–output connection. Here, we have presented a scheme for determining whether a given permutation is realizable in the GSE in a single pass.*

1. Introduction

Multistage interconnection networks (MINs) of size $N \times N$, where $N = 2^n$, for $n > 1$ have been known for a long time. There exists a wide class of MINs, e.g. omega, delta, baseline, reverse baseline, banyan, etc., which are known as full-access unique-path interconnection networks (Wu and Feng 1980, 1981), i.e. there exists a unique path between any input–output pair. Some have $\log_2 N$ stages for N inputs and are blocking by nature, whereas Benes and $(2 \log_2 N - 1)$-stage shuffle-exchange networks are examples of rearrangeable MINs which can realize all possible permutations of input–output connections (Abdennadher and Feng 1992); some rearrangements of existing connections may be required to accommodate some new paths.

The conventional full-access unique-path MINs are designed for $N = 2^n$ only. Thus, when the number of inputs N is not a power of 2, we have to go for a MIN of size $2^{\lceil \log N \rceil} \times 2^{\lceil \log N \rceil}$ and this causes wastage of resources as there are $2^{\lceil \log N \rceil} - N$ extra links per stage. For example, with $N = 1030$, $1024 < N < 2048$, we are to use a conventional full-access MIN with 11 ($\lceil \log N \rceil$) stages, the number of switches per stage will be 1024, with 2048 links per stage. Among these only 1030 links, equivalently 515 switches will be used at a time. Therefore, an appreciable amount of resources will remain unutilized.

In this paper, we take the idea of perfect shuffle, as can be done on a deck of cards, to propose a new $N \times N$ multistage interconnection network called GSE (Generalized Shuffle Exchange). In fact perfect shuffle is a well-known form of interconnection in multiprocessor systems, first introduced by Stone (1971). Given a vector of N elements, the perfect shuffle of this vector causes a permutation, elements of the first half of the vector are interlaced with the elements of the second half. This interconnection has been found not only to be useful for particular algorithms but to have a wide variety of applications. So far, N has been assumed to be a power of 2 only. In this paper, we apply the concept of perfect shuffle to construct the generalized shuffle

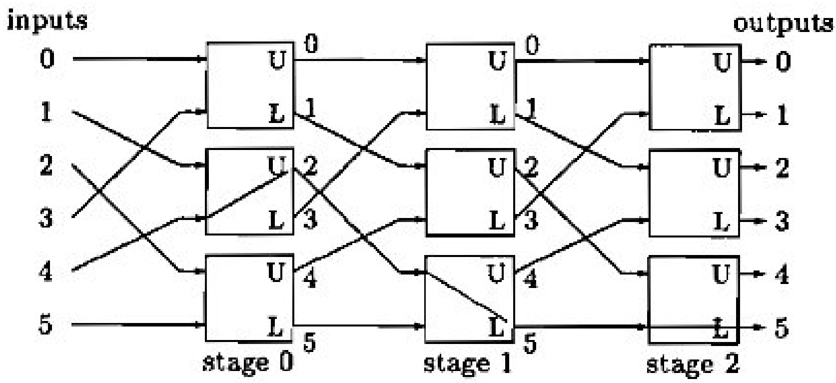

Figure 1. 6 × 6 GSE

exchange network (GSE), where N can be any even integer (as we require $N/2$ switches per stage, we need not consider the case when N is odd). If $2^{n-1} < N < 2^n$, the GSE will have n stages which is the minimum number of stages required for full-accessibility. With only $N/2$ switches per stage instead of 2^{n-1}, the GSE will be cheaper by a factor of N/N_1, where $N_1 = 2^n = 2^{\lceil \log N \rceil}$.

We show that GSE is a full-access network, and destination tag routeing can be applied as in a conventional shuffle–exchange (SE) network subject to a simple transformation on the destination address. We have then developed an algorithm for routeing in GSE. For a given input x, we have identified the set of outputs for which there exist alternative paths. Now, given an arbitrary permutation, the problem of determining whether the permutation is realizable by the network in a single pass or not is referred to as the permutation admissibility problem (Shen *et al.* 1995, Shen 1995). Here, we have outlined a scheme for resolving the permutation admissibility problem on a GSE.

The rest of the paper is organized as follows. Section 2 presents the structure and some properties of the GSE. In §3, we prove that GSE is full-access and gives an algorithm for routeing. In §4, we analyse the permutation admissibility problem and show how it can be resolved.

Finally, in §5, we make some concluding remarks and discuss the scope for future work.

2. Generalized shuffle exchange (GSE) network

We give an example of a 6 × 6 GSE in Fig. 1. The inputs, outputs and the output links of each stage are labelled as shown in the figure. Here, the number of stages is equal to $\lceil \log 6 \rceil = 3$ and the total number of switches is equal to 9. Compare this with the total number of switches in an 8 × 8 shuffle-exchange (SE) network, which is equal to 12. Note that each stage of GSE involves a shuffle and switches are used for the exchange.

The shuffle operation can be formally defined as follows.

Definition 1: A shuffle can be represented by a function

$$sh : \{0, 1, \ldots, (N-1)\} \rightarrow \{0, 1, \ldots, N-1\},$$

where

$$sh(x) = \begin{cases} 2x, & \text{if } x < \dfrac{N}{2} \\ 2x - N + 1, & \text{if } x \geq \dfrac{N}{2}. \end{cases} \qquad \square$$

Depending upon the state of the switch (crossed/straight) there may or may not be an exchange. The operation of a switch can be formally defined as:

$$sw : \{2x, 2x+1\} \rightarrow \{2x, 2x+1\},$$

where

$$sw(y) = \begin{cases} y, & \text{if no exchange takes place,} \\ y+1, & \text{if exchange and } y \text{ is even,} \\ y-1, & \text{if exchange and } y \text{ is odd.} \end{cases}$$

Of the two output links of a switch, we call the even numbered link the *upper output* and the odd numbered link the *lower output*.

Our definition of shuffle and exchange also applies to the case when N is a power of 2, i.e. the conventional $\log N$ stage SE network is a special case of GSE.

There follows here a set of notations and definitions to be used in the analysis of the GSE network:

x	source node/input, $x \in \{0, 1, \ldots, N-1\}$,
y	destination node/output, $y \in \{0, 1, \ldots, N-1\}$,
n	number of stages $= \lceil \log_2 N \rceil$,
Stage 0	first stage from the input side,
Stage $n-1$	last stage from the input side.

We give a routeing algorithm to connect any input x to any output y. In such routeing we only need to specify at each stage whether the lower or upper output will be used, i.e. the path can be uniquely specified by a routeing vector $R = r_{n-1}r_{n-2}\cdots r_1r_0$, $r_i \in \{0, 1\}$, where

$$r_i = \begin{cases} 0, & \text{if the upper output of the switch at } n-1-i \text{ is used} \\ 1, & \text{if the lower output of the switch at } n-1-i \text{ is used.} \end{cases}$$

Example 1: A path $4 \to 5$, i.e. from input 4 to output 5, will have the routeing vector $R = 011$, as has been shown in Fig. 1. The bits in R determine which output link (upper or lower) should be followed. Note that for the path $1 \to 5$ the routeing vector is also the same (Fig. 1). □

Under this definition of R it is immaterial whether the i put to the switch is lower or upper. Hence, we modify e function sh and define a function s called shift as llows.

efinition 2: A shift can be represented by a function

$$s : \{0, 1, \ldots, (N-1)\} \to \{0, 2, 4, \ldots, N-2\}$$

iere

$$s(x) = \begin{cases} 2x, & \text{if } x < \dfrac{N}{2} \\ 2x - N, & \text{if } x \geq \dfrac{N}{2}. \end{cases}$$ □

ote that $s(x)$ is always even, i.e. as if the outcome of a uffle always goes to the upper input of a switch. $s(x)$ n also be expressed as $s(x) = 2x \bmod N$.

The function of a switch is represented by two func- ons g and I defined as follows.

efinition 3:

$$g : \{0, 2, 4, \ldots, N-2\} \to \{1, 3, 5, \ldots, N-3, N-1\}$$
$$I : \{0, 2, 4, \ldots, N-2\} \to \{0, 2, 4, \ldots, N-2\}$$

vhere $g(x) = x + 1$ and $I(x) = x$. □

As $g(x)$ is an odd number, applying g implies that a lower output of the switch is used. I is an identity operation and $I(x)$ is always even. Thus, I corresponds to the upper output of a switch.

3. Routeing in GSE

In this section, we show that a path between any input x to any output y can be set up as follows.

(i) Apply a simple transformation on the destination y to get y'.

(ii) Use y' as the routeing vector to get the path.

3.1. *Routeing vectors in GSE*

From the previous discussion it follows that corresponding to a routeing vector R there is a path $P(R)$ which is a sequence $st_0st_1\cdots st_{n-2}st_{n-1}$, where $t_i \in \{g, I\}$ indicates the operation of the switch in the ith stage and

$$t_i = \begin{cases} g & \text{if } r_{n-1-i} = 1, \\ I & \text{if } r_{n-1-i} = 0. \end{cases}$$

Now, if the path P sonnects the source node 0 to a destination node y, then P and y are related by a simple relation which is stated in the following lemma.

Lemma 1: *$P(R)$ is a path between input 0 and output y if $R = y$.*

Proof: Let $R = r_{n-1}r_{n-2}\cdots r_1r_0$. As defined before, $P(R)$ is a sequence $st_0st_1s\cdots st_{n-2}st_{n-1}$, where

$$t_i = \begin{cases} g & \text{if } r_{n-1-i} = 1 \\ I & \text{if } r_{n-1-i} = 0. \end{cases}$$

From the definition of g and I, if we operate $P(R)$ on the source node $0 = 00\cdots 0$, then we will get the following sequence:

$$0 \to (s(0) + r_{n-1}) \to (s^2(0) + sr_{n-1} + r_{n-2} \to \cdots$$
$$\to (s^n(0) + s^{n-1}r_{n-1} + s^{n-2}r_{n-2}\cdots + sr_1 + r_0)$$

where s^i denotes i successive shift operations. Since, r_i are either 0 or 1, $s^k(r_i) = 2^k(r_i)$. Hence, the path P connects the input 0 to output $r_{n-1}r_{n-2}\cdots r_1r_0 = R = y$. □

Corollary 1: *Destination tag routeing is possible when the input is 0 as the routeing vector is simply the n-bit binary representation of the output y.*

For routeing between any input x ($x \neq 0$) and some output y, we claim that the routeing vector can be generated by applying a simple transformation on y.

Lemma 2: *A path $P(R)$ connects input x to output y, if $R = y - s^n(x)$ (mod N).*

Proof: Let $R = r_{n-1}r_{n-2}\cdots r_1r_0$.

We can write the sequence of links through which the path P goes as follows:

$$x \to (s(x) + r_{n-1}) \to (s^2(x) + sr_{n-1} + r_{n-2}) \to \cdots$$
$$\to (s^n(x) + s^{n-1}r_{n-1} + s^{n-2}r_{n-2} + \cdots sr_1 + r_0),$$

where all numbers are modulo N.

Thus, the operation P finally leads to the link $s^n(x) + R \pmod N = s^n(x) + y - s^n(x) \pmod N = y$. □

Corollary 2: *The routeing vector for connecting x and y is the same as the routeing vector for connecting 0 and y' where $y' = y - s^n(x)$ (mod N).*

We have defined $s(x) = 2x \mod N$. Hence, $s^n(x) = (2^n x) \mod N$ and for an input–output pair (x, y), routeing vector R is equal to $(y - 2^n(x)) \mod N$.

Theorem 1: *GSE is a full-access network.*

Proof: The proof follows directly from Lemmas 1 and 2. □

.2. *Alternative paths in GSE*

For an $N \times N$ GSE, $2^{n-1} < N < 2^n$, the routeing vector can take any of the 2^n values but there are only $N < 2^n$ destinations. So, given a source node x there must be some destination nodes which are routeable by more than one different routeing vector. Now we give the following lemma.

Lemma 3: *Given a routeing vector R', $N \le R' \le 2^n - 1$, if $R = R' - N$ routes from input x to output y, then R' routes from x to y. Also, the two paths corresponding to R and R' are disjoint.*

Proof: We have $N \le R' \le 2^n - 1$, which implies $0 \le R' - N \le 2^n - N - 1$. By Lemma 2, $y = s^n(x) + R' - N \pmod N$. Hence, $y = s^n(x) + R' \pmod N$, which proves the first part of the lemma.

Let $R = r_{n-1}r_{n-2}\cdots r_1 r_0$ and $R' = r'_{n-1}r'_{n-2}\ldots r'_1 r'_0$. Also let the path corresponding to R be

$$x \to a_1 \to a_2 \cdots a_{n-1} \to a_n = y, \quad a_i \in \{0, 1, \ldots, N-1\}$$

and the path corresponding to R' be

$$x \to b_1 \to b_2 \cdots b_{n-1} \to b_n = y, \quad b_i \in \{0, 1, \ldots, N-1\}$$

To show that these two paths are disjoint we have to show that $a_i \ne b_i$ for all i. Now,

$$b_i = s^i(x) + r'_{n-1}r'_{n-2}\cdots r'_{n-i}$$

and

$$a_i = s^i(x) + r_{n-1}r_{n-2}\cdots r_{n-i}$$

But,

$$r'_{n-1}r'_{n-2}\cdots r'_{n-i} \ne r_{n-1}r_{n-2}\cdots r_{n-i},$$

because, otherwise

$$r'_{n-1}r'_{n-2}\cdots r'_{n-i} - r_{n-1}r_{n-2}\cdots r_{n-i} = N$$

for some $i \le n-1$ and since we need at least n bits to represent N, this is not possible. Hence, the proof. □

Corollary 3: *Two disjoint paths corresponding to R and R' do not pass through any common switch (except the switches at stage-0 and stage-$(n-1)$).*

Proof: a_i, b_i are the output links of the switches $\lfloor a_i/2 \rfloor$ and $\lfloor b_i/2 \rfloor$. It can be shown as before that for all i, $0 \le i \le n-1$, $\lfloor a_i/2 \rfloor \ne \lfloor b_i/2 \rfloor$. □

Remark 1: Given an input x, there are $2^n - N$ destinations of the form $(s^n(x) + j) \mod N$, $0 \le j \le 2^n - N - 1$, for which there are two disjoint paths. For the remaining $2N - 2^n$, destinations which are of the form $(s^n(x) + j) \mod N$, $2^n - N \le j \le N - 1$, there is only a single path. □

For each source, we can draw a reachability tree showing the paths from source to all the destinations. A reachability tree for $N = 6$ and source node 0 is shown in Fig. 2. Here, the root node is the source, the leaves are the destinations and intermediate nodes are switching elements. The edges from upper and lower outputs from a switch are represented by two links labeled u and l respectively. Also, the reachability tree for some source node x, $x \le N/2$, is the same as the reachability tree for the source node $(x + N/2)$, as $s(x) = s(x + N/2)$ (actually they are the inputs to the same switch in the first stage). Note that there are alternative paths for destinations 0 and 1 from the source 0.

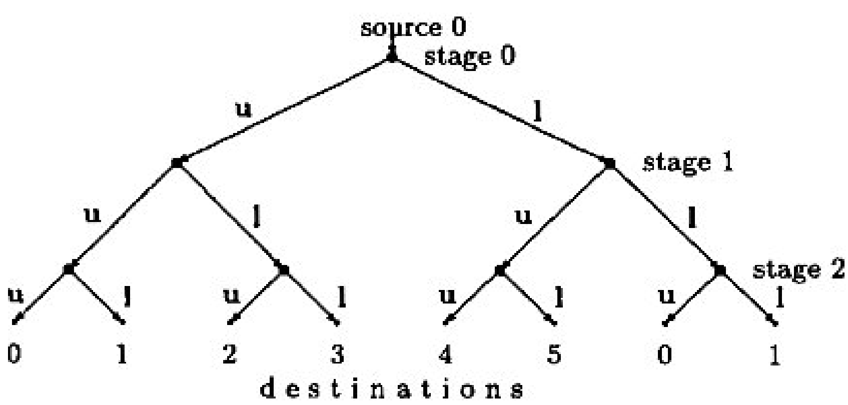

Figure 2. Reachability tree with source node 0

4. Permutation admissibility of GSE

An $N \times N$ MIN ($N = 2^n$) with $\log_2 N$ stages is a blocking network, because two paths for two source-destination pairs may demand the same link and therefore cannot be established simultaneously. This is called conflict.

Definition 4: An $N \times N$ permutation is said to be *admissible* to a MIN, if N conflict-free paths, one for each source-destination pair can be set up simultaneously through the MIN. □

Determining whether a given permutation can be realized by a MIN in a single pass is referred to as the permutation admissibility (PA) problem. An $O(N \log N)$ algorithm for the PA problem on an $N \times N$ MIN ($N = 2^n$) with $\log N$ stages has been developed by Shen *et al.* (1995). In Shen (1995), this work has been extended to k-extra stage cube type (k-EMCTN) MINs and it has been shown that a given permutation is admissible in k-EMCTN, if and only if its conflict graph is 2^k colourable.

We now present a scheme for resolving the PA problem in GSE. Given an input–output pair (x, y), there may exist either one or two paths in the GSE. This can be found as follows. We first find a routeing vector $R = (y - 2^n(x)) \bmod N$. If $0 \le R \le 2^n - N - 1$, then there are two disjoint paths with routeing vectors R and $R' = R + N$. But if $2^n - N \leqslant R \leqslant N - 1$, then there is a single path.

The links in each stage are numbered from 0 to $N - 1$ and we can identify a path from x to y by a sequence of links $l_0 l_1 \cdots l_i \cdots l_{n-1}$, where $l_0 = x$ and l_i, $1 \le i \le n$ is the link in the path after a switch in stage $i - 1$. Now, for each source and routeing vector pair, we find the respective links used by the path in stage i, $(0 \le i < n)$ by the equation $l_{i+1} = (s(l_i) + r_{n-i-1}) \bmod N$.

Definition 5: Given an $N \times N$ permutation π, if there exist two paths for m source-destination pairs each, $(m \le N)$, we define an $(N + m) \times (n + 1)$ transition matrix T_π, in which each row represents a path as a sequence of links l_i used by the path, $0 \le i \le n$. □

Example 2: Consider a permutation $\pi = \begin{pmatrix} 0 & 1 & 2 & 3 & 4 & 5 \\ 0 & 5 & 3 & 1 & 2 & 4 \end{pmatrix}$.

Here, $N = 6$, $n = 3$, and $R = r_2 r_1 r_0$. The links used at different stages are shown in Table 1.

From this we construct the transition matrix T_π as shown below.

Table 1. All possible paths for permutation π

x	y	R/R'	$l_1 = s(x) + r_2$	$l_2 = s(l_1) + r_1$	$l_3 = s(l_2) + r_0$
0	0	0	0	0	0
		6	1	3	0
1	5	3	2	5	5
2	3	5	5	4	3
3	1	1	0	0	1
		7	1	3	1
4	2	0	2	4	2
		6	3	1	2
5	4	0	4	2	4
		6	5	5	4

$$T_\pi = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 1 & 3 & 0 \\ 1 & 2 & 5 & 5 \\ 2 & 5 & 4 & 3 \\ 3 & 0 & 0 & 1 \\ 3 & 1 & 3 & 1 \\ 4 & 2 & 4 & 2 \\ 4 & 3 & 1 & 2 \\ 5 & 4 & 2 & 4 \\ 5 & 5 & 5 & 4 \end{pmatrix}.$$

From this transition matrix T_π we try to construct an $N \times (n + 1)$ path matrix M which has only one row for each source destination pair such that in each column no two elements are identical. (Two identical elements in some column i indicates that two paths are using the same link at stage i and hence there is a conflict.) If, for any given T_π, such a path matrix M does not exist, π is not admissible. For example, we have been able to construct a path matrix M for the given T_π as shown.

$$M = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 1 & 2 & 5 & 5 \\ 2 & 5 & 4 & 3 \\ 3 & 1 & 3 & 1 \\ 4 & 3 & 1 & 2 \\ 5 & 4 & 2 & 4 \end{pmatrix}.$$

Here, in each column no two elements are identical and hence the permutation π is routeable by the paths represented by the rows of M.

Remark 2: A permutation π is admissible in GSE, if and only if we can select N rows from T, one for each source–destination pair, such that in each column no two elements are identical. □

5. Conclusions

We have removed the restriction on the size of a MIN and introduced $N \times N$ generalized shuffle-exchange networks with $\lceil \log N \rceil$ stages. It has been shown that GSE is a full-access MIN which can interconnect any number of processors at minimum cost (even when the number of inputs N is odd, the number of switches per stage has to be $\lceil N/2 \rceil$ which is the required number of switches with $N+1$ inputs). Routeing in GSE is simple and the routeing vector can be computed using the destination address in constant time, since for a path $x \rightarrow y$, the routeing vector $R = y - s^n(x) \pmod N$ (Lemma 2).

When N is not a power of 2, for some input–output pairs there exist alternative paths which are disjoint. This feature adds some fault-tolerant capability to the GSE and makes it more attractive.

The scope of future research on GSE includes the following.

(i) Permutation capability of GSE: the aim is to find the number of permutations realizable by GSE and to characterize such permutations.

(ii) To design a rearrangeable MIN using more than $\lceil \log N \rceil$ number of stages. It is known that when $N = 2^n$, the $2n-1$ stage SE network is rearrangeable (Abdennadher and Feng 1992), i.e. the network can realize any $N \times N$ permutation. When $2^{n-1} < N < 2^n$, the minimum number of stages required for rearrangeability is greater than $2(n-1)-1 = 2n-3$. Our conjecture is that there exists an integer N', $2^{n-1} < N' < 2^n$, such that for $2^{n-1} < N < N'$, the minimum number of stages required for rearrangeability is $2n-2$ and for $N' \leq N \leq 2^n$, the corresponding number is $2n-1$.

References

Abdennadher, A., and Feng, T., 1992, On rearrangeability of Omega-Omega networks, *Proceedings of the 1992 International Conference on Parallel Processing*, pp. I-159–165.

Shen, X., Xu, M., and Wang, X., 1995, An optimal algorithm for permutation admissibility to multistage interconnection networks. *IEEE Transactions on Computers*, **44**, 604–608.

Shen, X., 1995, An optimal $O(N \lg N)$ algorithm for permutation admissibility to extra-stage cube-type networks. *IEEE Transactions on Computers*, **44**, 1144–1149.

Stone, H. S., 1971, Parallel processing with the perfect shuffle. *IEEE Transactions on Computers*, **20**, 153–161.

Wu, C. L., and Feng, T., 1980, On a class of multistage interconnection networks. *IEEE Transactions on Computers*, **29**, 694–702; 1981, The universality of the shuffle-exchange network. *IEEE Transactions on Computers*, **30**, 324–332.