

## Simulating a Mesh with Separable Buses

SUSUMU MATSUMAE<sup>†</sup> and NOBUKI TOKURA<sup>†</sup>

This paper investigates the problem of simulating a *mesh with separable buses* (MSB) by a *mesh with partitioned buses* (MPB) and a *mesh with restricted separable buses* (MRSB). The MSB and the MPB are the two-dimensional mesh-connected computers which have additional broadcasting buses along every row and column. The broadcasting buses of the MSB can be dynamically sectioned into smaller bus segments of various lengths by the program control, while those of the MPB are statically partitioned in advance by a fixed length  $\ell$ . The MRSB is a restricted model of the MSB, in which the broadcasting buses are placed only every  $\ell$  rows and  $\ell$  columns, and only those processors located at the crossing point of the broadcasting buses can access to the buses. We show that the MSB of size  $n \times n$  can be simulated in  $\Theta(n^{1/3})$  time by the MPB of size  $n \times n$  when  $\ell = \Theta(n^{2/3})$ , and in  $\Theta(\ell)$  time by the MRSB of size  $n \times n$ . These time costs are shown to be optimal in the worst case.

### 1. Introduction

The mesh architecture has been studied as one of parallel computational models. A two-dimensional mesh-connected computer (mesh for short) is a processor array that consists of processors arranged in a two-dimensional grid. Each processor is connected via bidirectional unit-time communication links to its four adjacent processors. The mesh is a natural structure for solving many problems in matrix computations and image processing, and thus many algorithms have been designed for such problems on it. Further, this structure is suitable for VLSI implementation and allows a high degree of integration.

The main disadvantage of the mesh is that the time complexity of an algorithm on it is lower-bounded by its large diameter. To overcome this problem, the mesh has been enhanced with various types of broadcasting buses. Stout<sup>1)</sup> and Prasanna-Kumar, et al.<sup>2)</sup> proposed a *mesh with multiple buses* (MWMB), which has additional broadcasting buses along each row and column. Maeba, et al.<sup>3)</sup> proposed a *mesh with separable buses* (MSB), where they considered sectioning the broadcasting buses of the MWMB by inserting the processor-controlled switches into the buses. The broadcasting buses of the MSB, called *separable buses*, can be dynamically divided into smaller bus segments by the program control.

Serrano, et al.<sup>4)</sup> considered a *mesh with restricted separable buses* (MRSB). The MRSB is a restricted model of the MSB in which the separable buses are placed only every  $\ell$  rows and  $\ell$  columns, and only those processors located at the crossing point of the broadcasting buses can access to the buses. Recently, Chung<sup>5)</sup> and Maeba, et al.<sup>6)</sup> proposed a *mesh with partitioned buses* (MPB). Like the MWMB and the MSB, the MPB has broadcasting buses along every row and column. The broadcasting buses of the MPB have no sectioning switch inserted, but are partitioned in advance by a fixed length  $\ell$ .

In this paper, we consider simulating the MSB by the MPB and the MRSB. We show that the MSB of size  $n \times n$  can be simulated in  $\Theta(n^{1/3})$  time by the MPB of size  $n \times n$  when  $\ell = \Theta(n^{2/3})$ , and that the MSB of size  $n \times n$  can be simulated in  $\Theta(\ell)$  time by the MRSB of size  $n \times n$ . Furthermore, these time costs are shown to be optimal in the worst case. Our motivations to consider these simulations are as follows. From a theoretical point of view, since we have shown<sup>7),8)</sup> that the MSB of size  $n \times n$  can simulate the *reconfigurable mesh*<sup>9),10)</sup> (or *PARBS*, the processor array with reconfigurable bus systems) of size  $n \times n$  in  $\Theta(\log^2 n)$  time, we can show that any problem that can be solved in time  $T$  by the reconfigurable mesh of size  $n \times n$  can be solved in  $O(TT' \log^2 n)$  time by the MPB or the MRSB of size  $n \times n$ , where  $T'$  is the time cost of simulating the MSB of size  $n \times n$ . Since it has been argued<sup>11)</sup> that the reconfigurable mesh can be used as a univer-

<sup>†</sup> Department of Informatics and Mathematical Science, Graduate School of Engineering Science, Osaka University

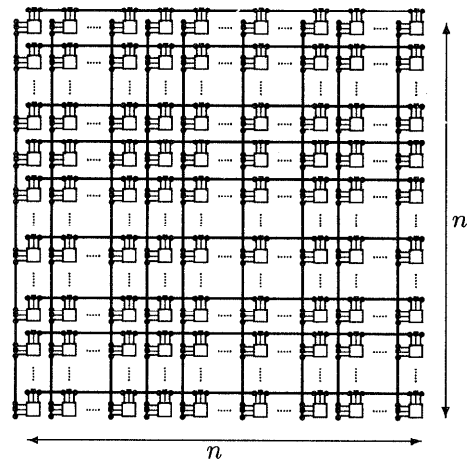
sal chip capable of simulating any equivalent-area architecture without loss in time, our simulation results give the upper bounds in time for the MPB and the MRSB to simulate other equivalent-area architectures. From a practical point of view, compared to the MSB, the number of broadcasting buses or that of switch elements inserted to the buses can be rather small on these simulating models, and thus it is expected that the scalability of them is better than that of the MSB. Also, considering the propagation delay of a broadcasting bus introduced by the length of the bus (i.e., signal propagation delay) and those switch elements inserted to the bus (i.e., device propagation delay), the propagation delay of the buses in the MPB and the MRSB can be small in practice, and hence we consider that our simulation algorithms are useful when the mesh size becomes so large that we cannot neglect the delay.

This paper is organized as follows. Section 2 describes the MSB, MPB, and MRSB models. Section 3 presents an algorithm that simulates the MSB on the MPB, and Section 4 gives an algorithm that simulates the MSB on the MRSB. And finally, Section 5 offers concluding remarks.

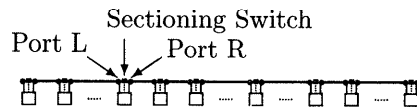
## 2. Models

An  $n \times n$  mesh consists of  $n^2$  identical SIMD processors or processing elements (PE's) arranged in a two-dimensional grid with  $n$  rows and  $n$  columns. The PE located at the grid point  $(i, j)$ , denoted as  $PE[i, j]$ , is connected via bidirectional unit-time communication links to those PE's at  $(i \pm 1, j)$  and  $(i, j \pm 1)$ , provided they exist ( $0 \leq i, j < n$ ).  $PE[0, 0]$  is located in the top-left corner of the mesh. Each  $PE[i, j]$  is assumed to know its coordinates  $(i, j)$ .

An  $n \times n$  mesh with separable buses (MSB) and an  $n \times n$  mesh with partitioned buses (MPB) are the  $n \times n$  meshes enhanced with broadcasting buses along each row and column (Fig. 1\* and Fig. 2). The broadcasting buses of the MSB, called separable buses, can be dynamically sectioned through the PE-controlled switches during the execution of programs, while those of the MPB are statically



(a) An  $n \times n$  MSB



(b) The broadcasting bus along a row of an  $n \times n$  MSB

Fig. 1 A Mesh with Separable Buses (MSB). Local links are not shown.

partitioned by a fixed length  $\ell$ . An  $n \times n$  mesh with restricted separable buses (MRSB) is the  $n \times n$  mesh enhanced with the separable broadcasting buses along every  $\ell$  rows and  $\ell$  columns for a fixed  $\ell$  (Fig. 3). In the MRSB, only  $PE[i\ell, j\ell]$  can access to the broadcasting buses ( $0 \leq i, j < n/\ell$ ).

A single time step of the MSB, MPB, and MRSB is composed of the following three substeps:

**Local Comm. Substep:** Every PE communicates with its adjacent PE's via local links.

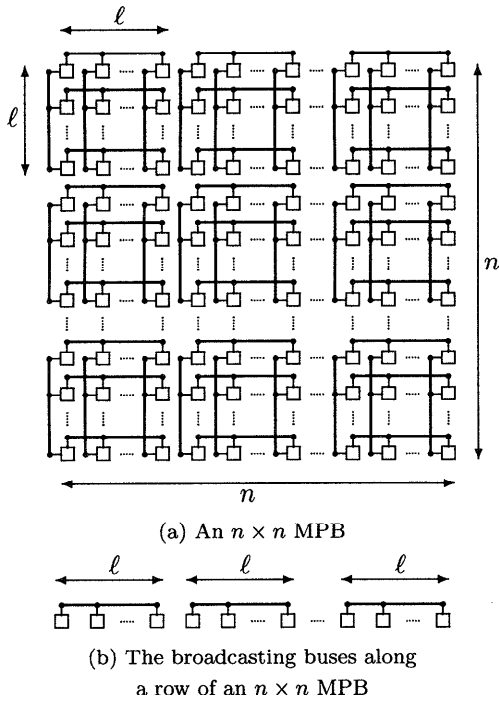
**Broadcast Substep:** Every PE changes its switch configurations by local decision (this operation is only for the MSB and the MRSB). After that, along each broadcasting bus segment, several of the PE's connected to the bus send data to the bus, and several of the PE's on the bus receive the data transmitted on the bus.

**Compute Substep:** Every PE executes some local computation.

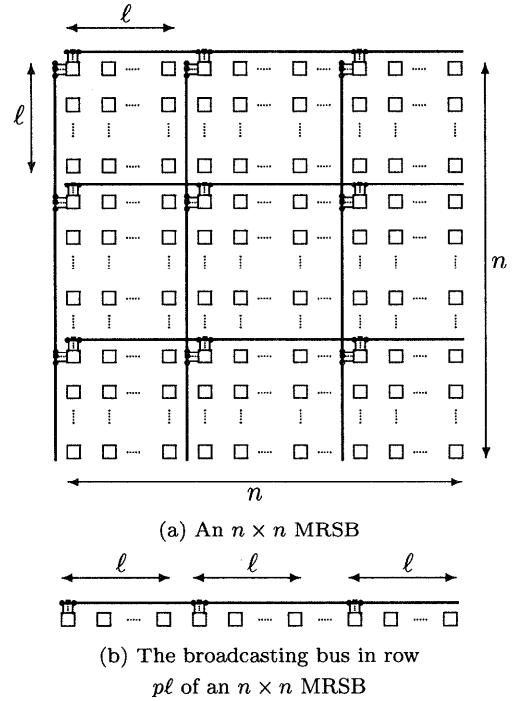
We assume that the propagation delay of the broadcasting buses is a constant time, and that each of the three substeps is executed in a constant time.

The bus accessing capability is similar to that

\* The MSB in this paper is slightly different from the one proposed by Maeba, et al.<sup>9)</sup>. Our model is closer (in fact, is equal in the computational power) to the submodel of the reconfigurable mesh, known as the horizontal-vertical reconfigurable mesh<sup>10)</sup>, in which every bus segment must be along row or column.



**Fig. 2** A Mesh with Partitioned Buses (MPB). Local links are not shown.



**Fig. 3** A Mesh with Restricted Separable Buses (MRSB). Local links are not shown.

of Common-CRCW PRAM model. If there is a write-conflict on a bus, the PE's on the bus receive a special value  $\perp$  (i.e., PE's can detect whether there is a write-conflict on a bus or not). If there is no data transmitted on a bus, the PE's on the bus receive a special value  $\phi$  (i.e., PE's can know whether there is data transmitted on a bus or not).

### 3. Simulation of MSB by MPB

In this section, we consider simulating a single step of the  $n \times n$  MSB (denoted as  $\mathcal{M}$ ) by the  $n \times n$  MPB (denoted as  $\mathcal{M}'$ ). To avoid confusion, let  $PE_{\mathcal{M}}[i, j]$  and  $PE_{\mathcal{M}'}[i, j]$  respectively denote  $PE[i, j]$  of  $\mathcal{M}$  and  $PE[i, j]$  of  $\mathcal{M}'$ .

Given a single step of  $\mathcal{M}$  in such a way that each  $PE_{\mathcal{M}'}[i, j]$  knows how  $PE_{\mathcal{M}}[i, j]$  behaves at this single step, we consider how to achieve the same computational task of the step using  $\mathcal{M}'$ . We assume that the computing power of PE's, the bandwidth of local links, and that of broadcasting buses are equivalent in both  $\mathcal{M}$  and  $\mathcal{M}'$ .

To begin with, we prove the following lemma.

**Lemma 1** For any single step of  $\mathcal{M}$ , the broadcasts taken on the separable bus in row  $i$  (resp. column  $i$ ) of  $\mathcal{M}$  can be simulated in row  $i$  (resp. column  $i$ ) of  $\mathcal{M}'$  in  $O(\ell + n/\ell)$  time

( $0 \leq i < n$ ).

**Proof:** Take any single step  $\mathcal{S}$  of  $\mathcal{M}$  and  $i \in \{0, 1, \dots, n-1\}$ . Let us consider simulating the broadcasts taken on the separable bus along row  $i$  of  $\mathcal{M}$  only, those on the bus along column  $i$  of  $\mathcal{M}$  can be simulated similarly.

First, we define some notations to describe the broadcasts to be simulated. Let  $P_j$  denote  $PE_{\mathcal{M}}[i, j]$  ( $0 \leq j < n$ ). To distinguish the two ports through which a PE has access to the row separable bus, we refer to the port on the left side of the sectioning switch as *port L* and the other as *port R*, as shown in Fig. 1 (b). Then, the broadcasts is carried out in the following way: (1) several of  $P_0, P_1, \dots, P_{n-1}$  section the bus, (2) several of  $P_0, P_1, \dots, P_{n-1}$  send data to the bus through port L and/or R, and (3) several of  $P_0, P_1, \dots, P_{n-1}$  receive data from the bus through port L and/or R. W.r.t. these broadcasts performed in row  $i$  of  $\mathcal{M}$ , we define  $C_j^x$ ,  $s_j^x$ , and  $r_j^x$  ( $0 \leq j < n$ ,  $x \in \{L, R\}$ ) as follows:

- $C_j^x = \{(k, y) \mid \text{port } x \text{ of } P_j \text{ and port } y \text{ of } P_k \text{ belong to the same bus segment after the broadcasting bus being sectioned}\}$ .
- $s_j^x = a$  if  $P_j$  sends data  $a$  to port  $x$ , otherwise  $s_j^x = \phi$ .

**Algorithm SB-by-PB**

{*Simulating the broadcasts taken on the separable bus in row  $i$  of  $\mathcal{M}$ , using row  $i$  of  $\mathcal{M}'$ .  $D_L, D_R, \mathbf{t}1, \mathbf{t}2, \mathbf{t}3$ , and  $\mathbf{t}4$  are the local variables in each PE.}*

**Phase 1:** {*Local Simulation*}

In each  $\mathcal{M}'_p$ , by sequentially scanning those  $s_j^L, s_j^R$ , and the switch configuration taken by  $P_j$  ( $pl \leq j < (p+1)\ell$ ) stored in the PE's of  $\mathcal{M}'_p$  from left to right and then from right to left, each  $P'_j$  obtains  $r_j^L$  and  $r_j^R$  in  $D_L$  and  $D_R$ , and knows whether  $(j, x) \in C_{l_p}^L$  and  $(j, x) \in C_{r_p}^R$  hold for each  $x \in \{L, R\}$  where  $l_p$  and  $r_p$  are the column indexes of  $LP'_{\mathcal{M}'_p}$  and  $RP'_{\mathcal{M}'_p}$ .

**Phase 2:** {*Global Simulation*}

for  $p \leftarrow 0$  to  $(n/\ell - 2)$  do

- (1)  $RP'_{\mathcal{M}'_p}$  sends  $D_R$  to  $LP'_{\mathcal{M}'_{p+1}}$ . The received data is stored in  $\mathbf{t}1$ .
- (2)  $LP'_{\mathcal{M}'_{p+1}}$  broadcasts  $\mathbf{t}1 \oplus D_L$  to all PE's in  $\mathcal{M}'_{p+1}$ . The received value is stored in  $\mathbf{t}2$  of each PE.
- (3) Each  $P'_j$  in  $\mathcal{M}'_{p+1}$  does  $D_x \leftarrow \mathbf{t}2$  if  $(j, x) \in C_{(p+1)\ell}^L$  ( $x \in \{L, R\}$ ).

for  $p \leftarrow (n/\ell - 1)$  to 1 do

- (1)  $LP'_{\mathcal{M}'_p}$  sends  $D_L$  to  $RP'_{\mathcal{M}'_{p-1}}$ . The received data is stored in  $\mathbf{t}3$ .
- (2)  $RP'_{\mathcal{M}'_{p-1}}$  broadcasts  $\mathbf{t}3$  to all PE's in  $\mathcal{M}'_{p-1}$ . The received data is stored in  $\mathbf{t}4$  of each PE.
- (3) Each  $P'_j$  in  $\mathcal{M}'_{p-1}$  does  $D_x \leftarrow \mathbf{t}4$  if  $(j, x) \in C_{pl-1}^R$  ( $x \in \{L, R\}$ ).

end of SB-by-PB

**Fig. 4** Algorithm SB-by-PB.

•  $r_j^x =$  (the data received by  $P_j$  from port  $x$ ).  
 To describe each  $r_j^x$  using  $C_j^*$  and  $s_j^*$ , we define a binary commutative operator  $\oplus$  in such a way that it satisfies the following equations for any  $x$  and  $y$ :

- $x \oplus \phi = \phi \oplus x = x$ ,
- $x \oplus \perp = \perp \oplus x = \perp$ ,
- $x \oplus y = y \oplus x = x$  if  $x = y$ ,
- $x \oplus y = y \oplus x = \perp$  if  $x \neq y, x \neq \phi$ , and  $y \neq \phi$ .

It is not difficult to confirm that  $\oplus$  is well-defined and enjoys the associative law. Then, each  $r_j^x$  can be expressed as:

$$r_j^x = \oplus_{(k,y) \in C_j^x} s_k^y. \tag{1}$$

Next, let  $P'_j$  denote  $PE_{\mathcal{M}'[i, j]}$  ( $0 \leq j < n$ ), and consider how to inform every  $P'_j$  of  $r_j^L$  and  $r_j^R$  when every  $P'_k$  is given  $s_k^L, s_k^R$ , and the switch configuration taken by  $P_k$ . We divide  $P'_0, P'_1, \dots, P'_{n-1}$  into  $n/\ell$  disjoint blocks  $\mathcal{M}'_p$  ( $0 \leq p < n/\ell$ ). Each  $\mathcal{M}'_p$  consists of  $P'_j$  ( $pl \leq j < (p+1)\ell$ ). Here, let  $LP'_{\mathcal{M}'_p}$  (resp.  $RP'_{\mathcal{M}'_p}$ ) denote the leftmost PE (resp. the rightmost PE) of  $\mathcal{M}'_p$ . It should be noted that  $RP'_{\mathcal{M}'_p}$  and  $LP'_{\mathcal{M}'_{p+1}}$  are adjacent PE's ( $0 \leq p < n/\ell - 1$ ) and that any PE in  $\mathcal{M}'_p$  can communicate with the other PE's in  $\mathcal{M}'_p$  in a single time step using the broadcasting bus ( $0 \leq p < n/\ell$ ). For each  $j \in \{0, \dots, n-1\}$  and  $x \in \{L, R\}$ , we let

$$r_j^x = \oplus_{(k,y) \in C_j^x} s_k^y \tag{2}$$

where  $C_j^x = C_j^x \cap \{(k, y) \mid P'_k \text{ and } P'_j \text{ are in the same block and } y \in \{L, R\}\}$ .

Then, in **Fig. 4**, we show an algorithm that simulates the broadcasts taken along row  $i$  of  $\mathcal{M}$ . Each  $r_j^x$  is stored in variable  $D_x$  of  $P'_j$  when the algorithm terminates. As for each  $r_j^x$  such that  $C_j^x \subset \{(k, y) \mid P'_k \text{ is in } \mathcal{M}'_p \text{ and } y \in \{L, R\}\}$  for some  $p \in \{0, \dots, n/\ell - 1\}$ , it is obtained at  $D_x$  of  $P'_j$  at Phase 1 because  $r_j^x = r_j^x$  holds in such a case. As for the rest, they are computed at Phase 2. After the execution of the first for-loop, such  $r_j^x$  is obtained at  $D_y$  of  $P'_k$  for every  $(k, y) \in C_j^x$  such that both  $pl \leq k$  and  $((p+1)\ell, L) \notin C_j^x$  hold for some  $p \in \{0, \dots, n/\ell - 1\}$ , and at the second for-loop, the value is copied to  $D_y$  of  $P'_k$  for every  $(k, y) \in C_j^x$ .

Phase 1 can be performed in  $O(\ell)$  time, since each block consists of  $\ell$  consecutive PE's. Note that this phase can be done similarly to the well-known algorithm on a linear processor array that performs a semigroup computation on values distributed one per processor by sequentially scanning those values. Phase 2 needs  $O(n/\ell)$  time. Thus, the conclusion follows.  $\square$

Next, we consider improving the time cost shown in Lemma 1. For Lemma 1, we presented the algorithm SB-by-PB in which the first phase

is performed in  $O(\ell)$  time by sequentially scanning data within each block. In the following, we reduce the time cost for this phase to  $O(\ell^{1/2})$  time, and as a result we obtain more efficient algorithm which runs in  $O(\ell^{1/2} + n/\ell)$  time.

As a corollary of Lemma 1, we state the following.

**Corollary 1** For any single step of  $\mathcal{M}$ , the broadcasts taken on the separable bus in row  $i$  (resp. column  $i$ ) of  $\mathcal{M}$  can be simulated in row  $i$  (resp. column  $i$ ) of  $\mathcal{M}'$  in  $O(n^{1/2})$  time when  $\ell = n^{1/2}$  ( $0 \leq i < n$ ).  $\square$

A close inspection of the algorithm used for proving Corollary 1 implies the following lemma<sup>\*</sup>.

**Lemma 2** For any single step of  $\mathcal{M}$ , the broadcasts taken on the separable bus in row  $i$  (resp. column  $i$ ) of  $\mathcal{M}$  can be simulated in row  $i$  (resp. column  $i$ ) of  $\mathcal{M}'$  in  $O(n^{1/2})$  time when  $\ell = n$  ( $0 \leq i < n$ ).

**Proof:** Consider the execution of SB-by-PB in a row of  $\mathcal{M}'$  with  $\ell = n^{1/2}$ . As for Phase 1, since broadcasting buses are not used for this phase, the length of the broadcasting bus of  $\mathcal{M}'$  doesn't matter. As for Phase 2, every time broadcast occurs, there is only a single PE sending a value in the row, and thus this phase can be performed even if there is only one broadcasting bus covering the entire row of  $\mathcal{M}'$ . Hence, every operation in Phases 1 and 2 can be executed in the same time on the  $n \times n$  MPB even when  $\ell = n$ . The proof for the simulation in each column is similar. Thus the conclusion follows.  $\square$

Then, we can now improve the result of Lemma 1, as shown below.

**Lemma 3** For any single step of  $\mathcal{M}$ , the broadcasts taken on the separable bus in row  $i$  (resp. column  $i$ ) of  $\mathcal{M}$  can be simulated in row  $i$  (resp. column  $i$ ) of  $\mathcal{M}'$  in  $O(\ell^{1/2} + n/\ell)$  time ( $0 \leq i < n$ ).

**Proof:** Let us consider simulating the broadcasts taken on the separable bus in row  $i$  of  $\mathcal{M}$  for a given step  $\mathcal{S}$  of  $\mathcal{M}$  only. To prove that the simulation can be performed in  $O(\ell^{1/2} + n/\ell)$  time, it suffices to show that the same task of Phase 1 of SB-by-PB can be achieved in  $O(\ell^{1/2})$  time. Here, we define  $P_j, P'_j, C_j^x, s_j^x, r_j^x, \oplus, \mathcal{M}'_p, LP'\mathcal{M}'_p, RP'\mathcal{M}'_p$ , and  $r_j^{x*}$  in the same way as in the proof of Lemma 1.

At Phase 1 of SB-by-PB, each  $\mathcal{M}'_p$  locally simulates the broadcasts (i.e., computes  $r_j^{x*}$ ).

Since each  $\mathcal{M}'_p$  can be seen as a linear array composed of  $\ell$  consecutive PE's and a broadcasting bus, by executing the algorithm proving Lemma 2 within each  $\mathcal{M}'_p$ , every  $P'_j$  can know  $r_j^{L*}$  and  $r_j^{R*}$  in  $O(\ell^{1/2})$  time.

Next, consider letting each  $P'_j$  in  $\mathcal{M}'_p$  know whether  $(j, x) \in C_{l_p}^L$  holds for each  $x \in \{L, R\}$  where  $l_p$  is the column index of  $LP'\mathcal{M}'_p$ . Consider a broadcast operation in row  $i$  of  $\mathcal{M}$  such that the bus configuration is the same as that of  $\mathcal{S}$  and every  $P_j$  corresponding to  $LP'\mathcal{M}'_p$  for some  $p$  sends "1" to the port  $L$ . Then, by locally simulating this broadcast operation within each  $\mathcal{M}'_p$ , every  $P'_j$  in  $\mathcal{M}'_p$  can know whether  $(j, x) \in C_{l_p}^L$  holds for each  $x \in \{L, R\}$  where  $l_p$  is the column index of  $LP'\mathcal{M}'_p$ . (Here, note that if  $P'_j$  obtains "1" for port  $X$  of  $P_j$  in this local simulation, it means that in the bus configuration of  $\mathcal{S}$  the port  $X$  of  $P_j$  is connected to the port  $L$  of  $P_k$  corresponding to  $LP'\mathcal{M}'_p$ .) By the similar argument in the preceding paragraph, this local simulation can be performed in  $O(\ell^{1/2})$  time in each  $\mathcal{M}'_p$ . Similarly, in  $O(\ell^{1/2})$  time, every  $P'_j$  in  $\mathcal{M}'_p$  can know whether  $(j, x) \in C_{r_p}^R$  holds for each  $x \in \{L, R\}$  where  $r_p$  is the column index of  $RP'\mathcal{M}'_p$ .

Thus, the same computational task of Phase 1 of SB-by-PB can be done more efficiently in  $O(\ell^{1/2})$  time. Since Phase 2 of SB-by-PB needs  $O(n/\ell)$  time, the entire simulation can be completed in  $O(\ell^{1/2} + n/\ell)$  time. Thus, the conclusion follows.  $\square$

In the proof of Lemma 3, we improved the time cost required for the Phase 1 of SB-by-PB by using the result of Lemma 2. Such an improvement is possible because the broadcasting buses of  $\mathcal{M}'$  are partitioned by length  $\ell$  and the algorithm proving Lemma 2 can be executed within each block in parallel. Because of this reason, we cannot apply the same speedup technique to the algorithm proving Lemma 3 in Ref. 7), though it is carried out in the same fashion as SB-by-PB.

Then, we obtain the following lemma immediately from Lemma 3.

**Lemma 4**  $\mathcal{M}'$  can simulate any single step of  $\mathcal{M}$  in  $O(\ell^{1/2} + n/\ell)$  time.

**Proof:**  $\mathcal{M}'$  can simulate the broadcast sub-step of a single step of  $\mathcal{M}$ , by first simulating the broadcasts taken along rows in parallel in each row, and then simulating those along columns similarly. This takes  $O(\ell^{1/2} + n/\ell)$

<sup>\*</sup> This lemma corresponds to Lemma 3 in Ref. 7).

time from Lemma 3. As for the local comm. and compute substeps,  $\mathcal{M}'$  can simulate them in a constant time in each PE. Thus, the conclusion follows.  $\square$

Next, we consider the lower bounds for simulating  $\mathcal{M}$  by  $\mathcal{M}'$ .

**Lemma 5** There exists a single step of  $\mathcal{M}$  that takes  $\Omega(n/\ell)$  time to be simulated on  $\mathcal{M}'$ .

**Proof:** Consider the single step of  $\mathcal{M}$  in which  $\text{PE}_{\mathcal{M}}[0, 0]$  sends a value to  $\text{PE}_{\mathcal{M}}[0, n - 1]$ . It is obvious that this step must take  $\Omega(n/\ell)$  time to be simulated on  $\mathcal{M}'$ .  $\square$

**Lemma 6** There exists a single step of  $\mathcal{M}$  that takes  $\Omega(\ell^{1/2})$  time to be simulated on  $\mathcal{M}'$ .

**Proof:** Consider the single step  $\mathcal{S}$  of  $\mathcal{M}$  whose broadcast substep consists of the following operations (here,  $L$  is some positive integer such that  $L \leq n$  and  $n \bmod L = 0$ ):

- (1) Each  $\text{PE}_{\mathcal{M}}[i, j]$  divides the row broadcasting bus if  $(j \bmod L) = 0$ .
- (2) Each  $\text{PE}_{\mathcal{M}}[i, j]$  sends the content of variable  $\mathbf{a}$  to the row broadcasting bus through port R if  $(j \bmod L) = 0$ .
- (3) Each  $\text{PE}_{\mathcal{M}}[i, j]$  receives data from the row broadcasting bus through port L and stores it in variable  $\mathbf{b}$  if  $(j \bmod L) \neq 0$ .

Let us call the data broadcasted at this substep as *a-values*. Note that there are possibly  $n^2/L$  different a-values.

Take any algorithm  $\mathcal{A}$  that correctly simulates  $\mathcal{S}$  on  $\mathcal{M}'$ . Here, the simulation is carried out on the MPB model, and thus the simulating PE's can use only the local links and the statically partitioned broadcasting buses. Since each PE initially holding an a-value is different from those PE's which will receive the value, every a-value must be transmitted through the local links and/or the broadcasting buses during the simulation. With these observations, we count the necessary steps for  $\mathcal{A}$ , by considering the following two cases:

**(Case 1):** There exists an *a-value* transmitted only through local links.

**(Case 2):** There is no *a-value* transmitted only through local links.

In Case 1, since the distance between the PE initially holding the a-value and the most distant PE which will receive it is  $L - 1$ ,  $\mathcal{A}$  must take at least  $L - 1$  steps. On the other hand, in Case 2, since the total number of a-value is  $n^2/L$ , and the number of data which may be transmitted on the broadcasting buses is at most  $2n^2/\ell$  in a single time step,  $\mathcal{A}$  must take at least  $\ell/(2L)$  steps. Thus, by letting  $L = \ell^{1/2}$ , in

either case,  $\mathcal{A}$  needs  $\Omega(\ell^{1/2})$  steps<sup>\*</sup>. Thus the conclusion follows.  $\square$

Now, we can state the following theorem.

**Theorem 1** When  $\ell = \Theta(n^{2/3})$ ,  $\mathcal{M}'$  can simulate any single step of  $\mathcal{M}$  in  $O(n^{1/3})$  time. This time cost is optimal in the worst case.

**Proof:** From Lemma 4,  $\mathcal{M}'$  can simulate any single step of  $\mathcal{M}$  in  $O(n^{1/3})$  time when  $\ell = \Theta(n^{2/3})$ . The optimality is from Lemma 5 and 6, since there exists a single step of  $\mathcal{M}$  which cannot be simulated in  $O(n^{1/3})$  time by any algorithm if  $\ell \neq \Theta(n^{2/3})$  and there exists a single step of  $\mathcal{M}$  which must take  $\Omega(n^{1/3})$  time to be simulated when  $\ell = \Theta(n^{2/3})$ .  $\square$

#### 4. Simulation of MSB by MRSB

In this section, we consider simulating the  $n \times n$  MSB by the  $n \times n$  MRSB. Let  $\mathcal{M}$  denote the  $n \times n$  MSB, and let  $\mathcal{M}'$  the  $n \times n$  MRSB. As in Section 3, we write  $\text{PE}_{\mathcal{M}}[i, j]$  and  $\text{PE}_{\mathcal{M}'}[i, j]$  for denoting  $\text{PE}[i, j]$  of  $\mathcal{M}$  and  $\text{PE}[i, j]$  of  $\mathcal{M}'$  respectively, and assume that the computing power of PE's, the bandwidth of local links, and that of broadcasting buses are equivalent in both  $\mathcal{M}$  and  $\mathcal{M}'$ .

We begin by proving lemmas for simulating broadcasts of  $\mathcal{M}$  by  $\mathcal{M}'$ .

**Lemma 7** For any single step of  $\mathcal{M}$ , the broadcasts taken on the separable bus in row  $i\ell$  (resp. column  $i\ell$ ) of  $\mathcal{M}$  can be simulated in row  $i\ell$  (resp. column  $i\ell$ ) of  $\mathcal{M}'$  in  $O(\ell)$  time ( $0 \leq i < n/\ell$ ).

**Proof:** Take any single step  $\mathcal{S}$  of  $\mathcal{M}$  and  $i \in \{0, 1, \dots, n/\ell - 1\}$ . Let us consider simulating the broadcasts taken on the separable bus along row  $i\ell$  of  $\mathcal{M}$  only, those on the bus along column  $i\ell$  of  $\mathcal{M}$  can be simulated similarly.

Let  $P_j$  and  $P'_j$  denote  $\text{PE}_{\mathcal{M}}[i\ell, j]$  and  $\text{PE}_{\mathcal{M}'}[i\ell, j]$  respectively ( $0 \leq j < n$ ).  $C_j^x$ ,  $s_j^x$ ,  $r_j^x$ ,  $\oplus$ ,  $\mathcal{M}'_p$ ,  $\text{LP}'\mathcal{M}'_p$ ,  $\text{RP}'\mathcal{M}'_p$ , and  $r_j^x$  are defined in the same way as in the proof of Lemma 1. Then, in Fig. 5, we show an algorithm that simulates the broadcasts performed along row  $i\ell$  of  $\mathcal{M}$ . Each  $r_j^x$  is stored in variable  $D_x$  of  $P'_j$  when the algorithm terminates.

After the execution of (2-2) of Phase 2, for each  $p \in \{0, \dots, n/\ell - 1\}$ ,  $r_{l_p}^L$  and  $r_{r_p}^R$  are obtained respectively in  $D_L$  and  $D'_R$  of  $\text{LP}'\mathcal{M}'_p$  where  $l_p$  is the column index of  $\text{LP}'\mathcal{M}'_p$  and  $r_p$  is that of  $\text{RP}'\mathcal{M}'_p$ . Then, using these information,

<sup>\*</sup> The lowerbound  $\Omega(n^{1/2})$  for simulating the MSB by the MWMB presented in 7) 8) is derived from this, since the  $n \times n$  MWMB is the same as the  $n \times n$  MPB with  $\ell = n$ .

**Algorithm SB-by-RSB**

{*Simulating the broadcasts taken on the separable bus in row  $i\ell$  of  $\mathcal{M}$ , using row  $i\ell$  of  $\mathcal{M}'$ .  $D_L$ ,  $D_R$ , and  $D'_R$  are the local variables in each PE.*}

**Phase 1:** {*Local Simulation*}

(1-1) Execute the Phase 1 of SB-by-PB.

(1-2) In each  $\mathcal{M}'_p$ , the content of  $D_R$  of  $RP'_{\mathcal{M}'_p}$  is transferred to  $LP'_{\mathcal{M}'_p}$ . This value is stored in  $D'_R$  of  $LP'_{\mathcal{M}'_p}$ .

**Phase 2:** {*Global Simulation*}

(2-1) Each  $LP'_{\mathcal{M}'_p}$  divides the row bus if  $(p\ell, L) \notin C_{(p+1)\ell-1}^R$ .

(2-2) Each  $LP'_{\mathcal{M}'_p}$  sends the content of  $D_L$  and that of  $D'_R$  to the bus through port L and R respectively. The values received from port L and R are stored in  $D_L$  and  $D'_R$ .

**Phase 3:** {*Local Propagation*}

In each  $\mathcal{M}'_p$ , PE's update  $D_L$  and  $D_R$  appropriately using the information of  $D_L$  and  $D'_R$  of  $LP'_{\mathcal{M}'_p}$ .

**end of SB-by-RSB**

**Fig. 5** Algorithm SB-by-RSB.

**Algorithm SBs-by-RSBs**

{*Simulating the broadcasts taken along rows of  $\mathcal{M}$ , using  $\mathcal{M}'$ .*}

**Stage 1:** {*Local Simulation*}

Execute the Phase 1 of SB-by-RSB in each row in parallel.

**Stage 2:** {*Global Simulation*}

In each  $\mathcal{B}'_p$ , do the following:

for  $i \leftarrow 0$  to  $(\ell - 1)$  do

Execute the Phase 2 of SB-by-RSB for the row  $i$  of  $\mathcal{B}'_p$ .

**Stage 3:** {*Local Propagation*}

Execute the Phase 3 of SB-by-RSB in each row in parallel.

**end of SBs-by-RSBs**

**Fig. 6** Algorithm SBs-by-RSBs.

each PE can update its  $D_L$  and  $D_R$  appropriately at Phase 3. The correctness is straightforward and we omit the details. Phases 1 and 3 can be performed in  $O(\ell)$  time, since each block  $\mathcal{M}'_p$  consists of  $\ell$  consecutive PE's. Phase 2 takes  $O(1)$  time. Hence, the conclusion follows.  $\square$

**Lemma 8** For any single step of  $\mathcal{M}$ , the broadcasts taken on the separable buses along rows (columns) of  $\mathcal{M}$  can be simulated on  $\mathcal{M}'$  in  $O(\ell)$  time.

**Proof:** Take any single step  $S$  of  $\mathcal{M}$ . Let us consider simulating the row broadcasts only, the column broadcasts can be simulated similarly.

We divide  $\mathcal{M}'$  into  $n/\ell$  disjoint bands  $\mathcal{B}'_p$  ( $0 \leq p < n/\ell$ ). Each  $\mathcal{B}'_p$  consists of  $PE_{\mathcal{M}'[i, j]}$  ( $p\ell \leq i < (p+1)\ell$ ,  $0 \leq j < n$ ), i.e.,  $\mathcal{B}'_p$  contains row  $i$  of  $\mathcal{M}'$  ( $p\ell \leq i < (p+1)\ell$ ). The row  $i$  of  $\mathcal{B}'_p$  is the row  $p\ell + i$  of  $\mathcal{M}'$  ( $0 \leq i < \ell$ ,  $0 \leq p < n/\ell$ ). Then, in **Fig. 6**, we show an algorithm that simulates the broadcasts along rows of  $\mathcal{M}$ .

Stage 1 and Stage 3 can be performed in  $O(\ell)$

time from Lemma 7. As for Stage 2, it can be done in  $O(\ell)$  time in the following way. In each  $\mathcal{B}'_p$ , only the row 0 of  $\mathcal{B}'_p$  has a broadcasting bus (restricted separable bus) whereby Phase 2 of SB-by-RSB is performed. Hence, for each row  $i$  ( $\neq 0$ ) of  $\mathcal{B}'_p$ , the data necessary for the execution of Phase 2 of SB-by-RSB must be moved to row 0 of  $\mathcal{B}'_p$ , and after the data being processed, the result must be moved back to the row. These operations can be done by just shifting the data synchronously with each iteration of the for-loop. Thus, Phase 2 can be completed in  $O(\ell + \ell) = O(\ell)$  time. Thus, the conclusion follows.  $\square$

Then, we have the following lemma.

**Lemma 9**  $\mathcal{M}'$  can simulate any single step of  $\mathcal{M}$  in  $O(\ell)$  time.

**Proof:**  $\mathcal{M}'$  can simulate the broadcast sub-step of a single step of  $\mathcal{M}$ , by first simulating the broadcasts taken along rows, and then simulating those along columns. This takes  $O(\ell)$  time from Lemma 8. As for the local comm.

and compute substeps,  $\mathcal{M}'$  can simulate them in a constant time in each PE. Thus, the conclusion follows.  $\square$

The lower bound for simulating  $\mathcal{M}$  by  $\mathcal{M}'$  is given in the following lemma.

**Lemma 10** There exists a single step of  $\mathcal{M}$  that takes  $\Omega(\ell)$  time to be simulated on  $\mathcal{M}'$ .

**Proof:** Consider the single step of  $\mathcal{M}$  in which  $PE_{\mathcal{M}}[0, 0]$  sends a value to  $PE_{\mathcal{M}}[0, \ell/2]$ . It is obvious that this step must take  $\Omega(\ell)$  time to be simulated on  $\mathcal{M}'$ .  $\square$

Now, we obtain the following theorem.

**Theorem 2**  $\mathcal{M}'$  can simulate any single step of  $\mathcal{M}$  in  $O(\ell)$  time. This time cost is optimal in the worst case.

**Proof:** From Lemmas 9 and 10.  $\square$

### 5. Concluding Remarks

We have shown that the  $n \times n$  MSB can be simulated time-optimally in  $\Theta(n^{1/3})$  time by the  $n \times n$  MPB when  $\ell = \Theta(n^{2/3})$ . Since we have proved<sup>7),8)</sup> that the  $n \times n$  MSB can be simulated time-optimally in  $\Theta(n^{1/2})$  time by the  $n \times n$  MWMB (which is equal to the  $n \times n$  MPB with  $\ell = n$ ), our result shows that we can reduce the time cost from  $\Theta(n^{1/2})$  to  $\Theta(n^{1/3})$  without increasing the number of switch elements.

Also, we have proved that the  $n \times n$  MSB can be simulated time-optimally in  $\Theta(\ell)$  time by the  $n \times n$  MRSB. Since the number of the sectioning switches used in the  $n \times n$  MRSB is  $2n^2/\ell^2$ , our results shows that we can reduce the number of the switch elements used in the MSB by the factor of  $\ell^2$  with paying the extra time cost proportional to  $\ell$ .

Although we assumed that the propagation delay of the broadcasting buses was a constant time, we cannot neglect the influence of the delay when the mesh size becomes large. Here, as in Ref. 12), let us take the following assumptions:

- The propagation delay of a bus is proportional to the sum of its signal propagation delay and device propagation delay.
- The signal propagation delay of an  $x$ -length bus is  $O(x^\alpha)$  for some  $\alpha \geq 0$ .
- The device propagation delay of a bus is neglectable (Case 1), or is proportional to the number of the switch elements inserted to the bus (Case 2).

Then, in Table 1, we show the necessary time to perform any single step of the  $n \times n$  MSB.

As for the Case 1, we can see that the MPB

**Table 1** Time costs to perform a single step of the  $n \times n$  MSB when the propagation delay cannot be neglected. In Case 1 we can neglect the device propagation delay, and in Case 2 we cannot. Each mesh is of size  $n \times n$ , and the broadcasting buses of the MPB are partitioned with  $\ell = n^{2/3}$ .

models	time costs	
	Case 1	Case 2
MSB	$O(n^\alpha)$	$O(n^\alpha + n)$
MPB	$O(n^{1/3} \cdot n^{2\alpha/3})$	$O(n^{1/3} \cdot n^{2\alpha/3})$
MRSB	$O(\ell \cdot n^\alpha)$	$O(\ell \cdot (n^\alpha + n/\ell))$

can perform a single step of the MSB as efficiently as the MSB if  $\alpha = 1$ , and that it is even superior to the MSB if  $\alpha > 1$ . As for the Case 2, the MPB is equal to (when  $\alpha = 1$ ) or superior to (when  $\alpha \neq 1$ ) the MSB, and the MRSB has the same efficiency as the MSB if  $\ell \leq n^{1-\alpha}$ . But from a theoretical point of view, in these cases (except the MPB in Case 2 with  $\alpha < 1$ ) there is less advantage of augmenting ordinary meshes with broadcasting buses, for it takes  $\Omega(n)$  time to perform a single step of the  $n \times n$  MSB.

Finally, we note that the MPB and the MRSB are suitable for VLSI implementation. The  $n \times n$  MPB can be constructed from the  $\ell \times \ell$  MWMB's, by putting them with  $n/\ell$  rows and  $n/\ell$  columns and connecting each adjacent MWMB's with  $\ell$  local links. Similarly, the  $n \times n$  MRSB can be constructed from the  $\ell \times \ell$  ordinary meshes, by putting them with  $n/\ell$  rows and  $n/\ell$  columns, connecting each adjacent mesh with  $\ell$  local links (in practice, we can prove that these local links are not necessary for simulating the MSB in  $O(\ell)$  time), and placing  $2n/\ell$  restricted separable buses.

**Acknowledgments** The authors thank the anonymous referees for many helpful comments to improve the quality of the paper.

### References

- 1) Stout, Q.F.: Meshes with Multiple Buses, *Proc. 27th Annual Symposium of Foundations of Computer Science*, pp.264-273 (1986).
- 2) Prasanna-Kumar, V.K. and Raghavendra, C.S.: Array Processor with Multiple Broadcasting, *J. Parallel Distributed Computing*, Vol.4, pp.173-190 (1987).
- 3) Maeba, T., Tatsumi, S. and Sugaya, M.: Algorithms for Finding Maximum and Selecting Median on a Processor Array with Separable Global Buses, *IEICE Trans. A*, Vol.J72-A, No.6, pp.950-958 (1989).
- 4) Serrano, M.J. and Parhami, B.: Opti-



- mal Architectures and Algorithms for Mesh-Connected Parallel Computers with Separable Row/Column Buses, *IEEE Trans. Parallel and Distributed Systems*, Vol.4, No.10, pp.1073–1080 (1993).
- 5) Chung, K.L.: Prefix Computations on a Generalized Mesh-Connected Computer with Multiple Buses, *IEEE Trans. Parallel and Distributed Systems*, Vol.6, No.2, pp.196–199 (1995).
  - 6) Maeba, T., Sugaya, M., Tatsumi, S. and Abe, K.: Semigroup Computations on a Processor Array with Partitioned Buses, *IEICE Trans. A*, Vol.J80-A, No.2, pp.410–413 (1997).
  - 7) Matsumae, S. and Tokura, N.: Simulation Algorithms among Enhanced Mesh Models, *IEICE Trans. Inf. & Syst.* (to appear)
  - 8) Matsumae, S. and Tokura, N.: Simulation Algorithms among Enhanced Mesh Models, Technical Report 99-ICS-1, Dept. of Informatics and Mathematical Science, Graduate School of Engineering Science, Osaka University (1999).
  - 9) Wang, B. and Chen, G.: Constant Time Algorithms for the Transitive Closure and Some Related Graph Problems on Processor Arrays with Reconfigurable Bus Systems, *IEEE Trans. Parallel and Distributed Systems*, Vol.1, No.4, pp.500–507 (1990).
  - 10) Ben-Asher, Y., Gordon, D. and Schuster, A.: Efficient Self-Simulation Algorithms for Reconfigurable Arrays, *J. Parallel and Distributed Computing*, Vol.30, No.1, pp.1–22 (1995).
  - 11) Miller, R., Prasanna-Kumar, V. K., Reisis, D. and Stout, Q. F.: Meshes with Reconfigurable Buses, *Proc. fifth MIT Conference on Advanced Research in VLSI*, Boston, pp.163–178 (1988).
  - 12) Maeba, T., Sugaya, M., Tatsumi, S. and Abe,

K.: An Influence of Propagation Delays on the Computing Performance in a Processor Array with Separable Buses, *IEICE Trans. A*, Vol.J78-A, No.4, pp.523–526 (1995).

(Received November 24, 1998)

(Accepted July 1, 1999)



**Susumu Matsumae** received the M.E. degree from Osaka University in 1996. He is currently a Ph.D. candidate at Osaka University. His research interests include parallel algorithms and architectures, logic, and computational complexity. He is a member of IEEE.



**Nobuki Tokura** received the B.E., M.E., and Ph.D. degrees in electronics engineering from Osaka University in 1963, 1965 and 1968, respectively. Since 1968, he has been working at Faculty of Engineering Science, Osaka University. He is a professor of Graduate School of Engineering Science, Osaka University. His research interests include theory and methods of program, algorithms and applications of information technologies to education. He is a member of the Institute of Electronics, Information and Communication Engineers, Japan Society for Software Science and Technology, ACM and IEEE.