Optimal All-to-All Personalized Communication in All-Port Tori

Liu Gang, Gu Nai-jie, Bi Kun, Tu Kun, and Dong Wan-li

Abstract—All-to-all personalized communication, also known as complete exchange, is one of the most dense communication patterns in parallel computing. In this paper, we propose new indirect algorithms for complete exchange on all-port ring and torus. The new algorithms fully utilize all communication links and transmit messages along shortest paths to completely achieve the theoretical lower bounds on message transmission, which have not be achieved among other existing indirect algorithms. For 2D $r \times c$ ($r \leq c$) all-port torus, the algorithm has time complexities of optimal transmission cost and $O(c)$ message startup cost. In addition, the proposed algorithms accommodate non-power-of-two tori where the number of nodes in each dimension needs not be power-of-two or square. Finally, the algorithms are conceptually simple and symmetrical for every message and every node so that they can be easily implemented and achieve the optimum in practice.

Keywords—Complete exchange, collective communication, all-to-all personalized communication, parallel computing, wormhole routing, torus.

I. INTRODUCTION

DISTRIBUTED-memory multiprocessors are widely employed to solve large scale scientific and engineering problems. It is widely recognized that interprocessor communication is one of the main bottlenecks in increasing the performance of multiprocessors in which the processors are linked by an interconnection network. Nowadays, multicomputers have provided high performance and scalable *collective communica- tion* [1] pattern, which involves global data movement and global control among a group of processors and is supported by the Message Passing Interface (MPI). Among these patterns, all-to-all personalized communication, or simply complete exchange, is the most dense communication pattern which requires that each node sends a distinct message of the same size to each other processor. Numerous scientific and numerical applications exhibit the need of such communication patterns, such as matrix algorithms, fast Fourier transformation (FFT), and graph algorithms, and are used to evaluate the quality of interconnection networks.

The network considered in this paper is *torus*, which has a

simple, regular topology and a bounded node degree. Because torus possesses excellent scalability and satisfies the demand of high bandwidth and low latency, it has been adopted by commercial machines, such as IBM Bluegene, Cray T3D/T3E, and Intel Paragon.

Complete exchange problem has been extensively studied in the past decade. Some researches are developed on the cluster-based systems [2]. However, most of these researches are designed to handle the topological constraints of the underlying networks, such as tori [3 – 8], meshes [9, 10], hyper-cubes [11], and multistage networks [12]. General speaking, algorithms for complete exchange can be classified into two categories: *direct* or *indirect* (message-combining) approaches. For the direct algorithm [3, 4], each processor directly sends those messages to each of the destination processors. For the indirect algorithm [5 – 9], messages may be delivered to their destination indirectly via intermediate processors in which messages are combined to form a larger message. Although the direct approach can achieve optimality in transmission cost, the startup cost is very high [5]. The indirect approach uses message combining to obviously reduce message startup cost. However, the indirect approach is very hard to completely achieve the theoretical lower bound on message transmission since it incurs more traffic in the network. Thus, indirect approach favors short messages exchange, while direct approach favors long messages exchange. In tori or meshes, the indirect algorithms tend to be more efficient than the direct ones.

Both direct algorithms in [3, 4] achieved the lower bound on message transmission of 2^{3d-3} on $2^d \times 2^d$ torus, but they finish the complete exchange operation in 2^{3d-3} communication steps. To relieve the direct algorithm's problem, Tseng [5] proposed a diagonal-propagation scheme that achieves $O(2^{3d})$ transmis- sion time and $O(2^d)$ startup time. In [6], Suh proposed indirect algorithms using message combining on $2^d \times 2^d$ tori with time complexities of $O(d)$ due to message startup and $O(2^{3d})$ due to message transmission. However, the constant associated with the transmission time is relatively high and the effect of this is significant as the size of message is fairly large. Tseng [7] used a "gather-then-scatter" technique and enforced shortest paths in routing messages to achieve asymptotically optimal startup time. Suh[10] presented more efficient indirect multidimensional algorithms that the size of network needs not be power-of-two and square.

Existing indirect algorithms can not fully utilize all communication links so as to fail in achieving optimality in transmission time. In addition, we are not aware of any existing algorithms for complete exchange on all-port tori. However, the number of algorithms for all-to-all broadcast on all-port tori is quite a few, such as [13, 14].

In this paper, we propose new indirect algorithms for complete exchange on all-port ring and 2D torus. We use message combining to reduce the startup time, fully utilize all communication links, and send messages along shortest paths to minimize transmission time. Compared with other existing algorithms, the proposed algorithms have following features: (1) They completely achieve the theoretical lower bounds on message transmission; (2) They accommodate non-power-of-two tori where the number of nodes in each dimension needs not be power-of-two or square; (3) They are conceptually simple and symmetrical for every message and every node.

The rest of this paper is organized as follows. In the next section, we present basic system model. We propose the algorithm for all-port ring in Sect. 3, the algorithm for 2D all-port torus in Sect. 4. Performance analysis and comparison is given in Sect. 5. Finally, conclusions are drawn in Sect. 6.

II. System Model

In this paper, we consider multicomputers composed of nodes interconnected together by a torus topology. We assume the communication model in which each communication channel is *full-duplex* and each node has *all-port* capability (as opposed to the *one-port*). In other words, a node in the network can simultaneously send and receive messages on a channel, and at any time each node can exchange messages with all of its neighbors simultaneously. This assumption is used in several recently constructed multiprocessors in order to fully use all of the available bandwidth. Fig. 1 depicts the internal structure of a node in 4×4 all-port torus. Each node is composed of a processor, a router which determines the route of messages arriving, leaving and passing through the node, and a buffer matrix required for complete exchange to store the messages. In addition, there are four pairs of first-in-first-out buffers in each node for 2D all-port torus, while each input buffer associated with an input channel of the node and each output buffer associated with an output channel of the node. In our algorithms, we assume that a message proceeds only along one dimension at a time, which is called as dimension-order routing.
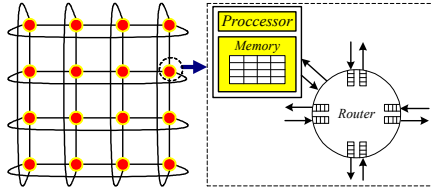

Fig. 1 The internal structure of a node in 4×4 all-port torus

We adopt wormhole routing technique in this paper, which is most popular switching technique. The communication latency in wormhole switching is almost independent of the number of hops between two nodes, if there is no contention in the channels. Let t_s be the startup time per message, which is the time required for the source node to prepare the message and initialize the communication, t_w be the message transmission time per byte, and ρ be the data rearrangement time per byte between communication phases. The communication time for one communication step can be expressed as $T = t_s + m \cdot t_w$, if one m-byte message is sent to the destination without any

contention. The following lemma gives the lower bound on message transmission for complete exchange operation in tori.

Lemma 1. For a k-dimensional torus of size $N_1 \times N_2 \times \cdots \times N_k$, where $N_1 \geq N_2 \geq \cdots \geq N_k$ and N_1 is even, the lower bound on message transmission for performing complete exchange is $N_1 \cdot \prod_{i=1}^{k} N_i \cdot m / 8$, where m is the size of per message.

Proof. Omitted. See [4] for details.

III. Basic Construct: Complete Exchange on an All-Port Ring

In this section, we consider the complete exchange problem on an all-port ring of p processors, where p is even and $p \geq 4$. Nodes on the ring are numbered clockwise from P_0 to P_{p-1}. For each node P_i, if $(j-i) \bmod p \leq p/2$, then we think node P_j ($j \neq i$) is on the *positive half-circle* of P_i, otherwise is on the *negative half-circle* of P_i. Each node P_s has a message denoted as M_d^s whose destination is node P_d. We use $M_d^{i,\dots,j}$ denote the set of messages $\{M_i^s, M_{i+1}^s, \dots, M_j^s\}$, and use $M_d^{i,\dots,j}$ to denote the set of messages $\{M_d^i, M_d^{i+1}, \dots, M_d^j\}$. If $i < 0$ or $i \geq p$, then we denote $(i \pm p) \bmod p$ as i.

We consider each bidirectional channel as two unidirectional channels, and consider a ring of p processors as two unidirectional sub-rings of $p/2$ processors: *odd sub-ring* and *even sub-ring*. The even sub-ring consists of only even-numbered nodes (simply even nodes) and unidirectional channels, and the odd sub-ring consists of odd-numbered nodes (simply odd nodes) and unidirectional channels. Each node on the ring can utilize its two ports to transmit messages along positive or negative direction simultaneously. The communication pattern is described as follows.

A. Communication Pattern

The algorithm has three stages. In Stage 1, each even node P_{2i} sends all messages whose destinations are odd nodes on the positive half-circle of P_{2i} to its positive adjacent odd node P_{2i+1} and sends all messages whose destinations are odd nodes on the negative half-circle of P_{2i} to its negative adjacent odd node P_{2i-1} simultaneously. At the same time, each odd node P_{2i+1} sends all messages whose destinations are even nodes on the positive half-circle of P_{2i+1} to its positive adjacent even node P_{2i+2} and sends all messages whose destinations are even nodes on the negative half-circle of P_{2i+1} to its negative adjacent even node P_{2i} simultaneously.

In Stage 2 and 3, complete exchange operations are performed in even sub-ring and odd sub-ring, respectively. In Stage 2, each node on the even sub-ring sends messages clockwise to the next even node, while each node on the odd sub-ring sends messages anticlockwise to the next odd node. Upon receiving the messages, each node extracts the messages meant for it and forwards the remainder to the next node in the direction of the messages. The process is repeated $\lfloor p/4 \rfloor$ times at most (i.e. half a circle) until all messages have reached their destination nodes. In Stage 3, each node on the even sub-ring sends messages anticlockwise to the next even node, while each node on the odd sub-ring sends messages clockwise to the next odd node. Similarly, the process is repeated $\lceil p/4 \rceil - 1$ times at

Algorithm *AR***:** // Complete Exchange on an All-Port Ring
BEGIN
{Stage 1}
For $i = 0$ To $p/2 - 1$ Para_Do

P_{2i} sends $M^{2i}_{2i+1,2i+3,\dots,2i+2\cdot\lfloor p/4\rfloor-1}$ to P_{2i+1}

P_{2i} sends $M^{2i}_{2i+2\cdot\lfloor p/4\rfloor+1,\dots,\,2i+p-3,2i+p-1}$ to P_{2i-1}

P_{2i+1} sends $M^{2i+1}_{2i+2,2i+4,\dots,2i+2\cdot\lceil p/4\rceil}$ to P_{2i+2}

P_{2i+1} sends $M^{2i+1}_{2i+2\times\lceil p/4\rceil+2,2i+2\times\lceil p/4\rceil+4,\dots,2i+p}$ to P_{2i}
Endfor
{Stage 2}
For $k = 0$ To $\lfloor p/4\rfloor - 1$ Do
 For $i = 0$ To $p/2 - 1$ Para_Do

P_{2i} sends $M^{2i-2k}_{2i+2,2i+4,\dots,\,2i+2\cdot\lfloor p/4\rfloor-2k}$ and $M^{2i-1-2k}_{2i+2,2i+4,\dots,\,2i+2\cdot(\lceil p/4\rceil-1)-2k}$ to P_{2i+2}

P_{2i+1} sends $M^{2i+1+2k}_{2i+1+2\cdot\lceil p/4\rceil+2k,\dots,\,2i+p-3,2i+p-1}$ and $M^{2i+2+2k}_{2i+1+2\cdot(\lfloor p/4\rfloor+1)+2k,\dots,2i+p-3,2i+p-1}$ to P_{2i-1}
 Endfor
Endfor
{Stage 3}
For $k = 0$ To $\lceil p/4\rceil - 2$
 For $i = 0$ To $p/2 - 1$ Para_Do

P_{2i} sends $M^{2i+2k}_{2i+2\cdot(\lfloor p/4\rfloor+1)+2k,\dots,2i+p-4,2i+p-2}$ and $M^{2i+1+2k}_{2i+2\cdot(\lceil p/4\rceil+1)+2k,\dots,2i+p-4,2i+p-2}$ to P_{2i-2}

P_{2i-1} sends $M^{2i-1-2k}_{2i+1,2i+3,\dots,\,2i-1+2\cdot(\lceil p/4\rceil-1)-2k}$ and $M^{2i-2-2k}_{2i+1,2i+3,\dots,\,2i-1+2\cdot(\lfloor p/4\rfloor-1)-2k}$ to P_{2i+1}
 Endfor
Endfor
END.

Fig. 2 Description of Complete Exchange on an All-Port Ring

(a)

(b)

(c)

(d)

All-Port Ring of 8 Nodes

most until all messages have reached their destination nodes. The formal description of complete exchange algorithm *AR* on all-port ring is shown in Fig. 2.

Fig. 3 illustrates the communication patterns for complete exchange on an all-port ring of eight nodes. The above algorithm guaranteed not only the absence of link contention but also the full utilization of the links and enforcement of shortest paths.

B. Complexity Analysis

In the following, we analyze the time complexity of algorithm *AR* in terms of startup time and message-transmission time.

Lemma 2: The total communication cost of our complete exchange scheme on an all-port ring of *p* nodes, where *p* is even and $p \geq 4$, is

$$T_{ring} = p/2 \cdot t_s + \lceil p^2/8 \rceil \cdot mt_w.$$

Proof. In Stage 1, each node exchanges messages with its two adjacent nodes simultaneously, therefore Stage 1 has only one step and $\lceil p/4 \rceil$ messages are transmitted. In Stage 2, messages are relayed $\lfloor p/4 \rfloor$ times at most on the even (odd) sub-ring, thus the transmission cost of Stage 2 is

$$\sum_{k=0}^{\lfloor p/4\rfloor-1}\left[\left(\lfloor p/4\rfloor-k\right)+\left(\lceil p/4\rceil-1-k\right)\right]\cdot mt_w \quad (1)$$
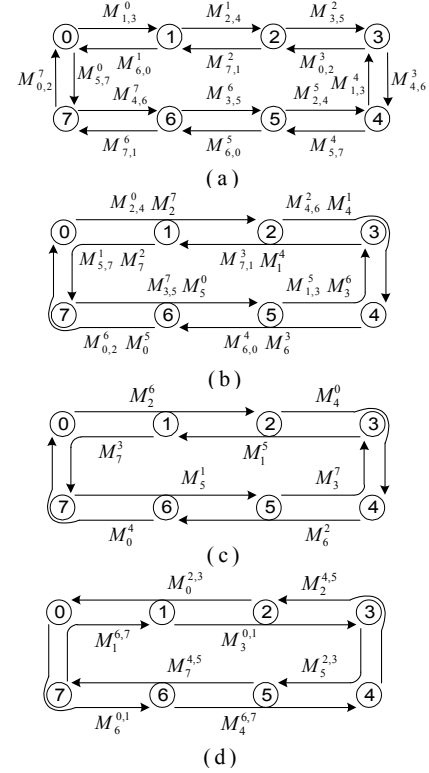$$=\lfloor p/4\rfloor\cdot\lceil p/4\rceil\cdot mt_w$$

Likewise, messages are relayed $\lceil p/4 \rceil - 1$ times at most on the odd (even) sub-ring in Stage 3, thus the transmission cost of Stage 3 is:

$$\sum_{k=0}^{\lceil p/4\rceil-2}\left[\left(\lceil p/4\rceil-1-k\right)+\left(\lfloor p/4\rfloor-1-k\right)\right]\cdot mt_w \quad (2)$$
$$=\lfloor p/4\rfloor\cdot\left(\lceil p/4\rceil-1\right)\cdot mt_w$$

Fig. 3 Complete Exchange on an

Thus, the time complexity of algorithm *AR* is $T_{ring} = p/2 \cdot t_s + \lceil p^2/8 \rceil \cdot mt_w$, where the message transmission time completely achieves the theoretical lower bound $\lceil p^2/8 \rceil \cdot mt_w$.

IV. COMPLETE EXCHANGE ON ALL-PORT 2D TORUS

In this section, we consider the complete exchange on 2D $r \times c$ all-port torus, where *r* and *c* are multiples of four and $r \leq c$. Each node in the torus is denoted as $P(x, y)$ ($0 \leq x \leq r-1$ and $0 \leq y \leq r-1$), which is connected to $P(x-1, y)$, $P(x+1, y)$, $P(x, y-1)$, and $P(x, y+1)$.

All nodes in a torus are topological symmetric. Without loss of generality, if we place node $P(x_0, y_0)$ at the center, every other node $P(x, y)$ in the torus is included into one of the center node's four quadrants, namely QI, QII, QIII, and QIV, according to the following rule:

QI: if $(x-x_0)\bmod r \leq r/2$ and $(y-y_0)\bmod c \leq c/2$;
QII: if $(x-x_0)\bmod r > r/2$ and $(y-y_0)\bmod c \leq c/2$;
QIII: if $(x-x_0)\bmod r > r/2$ and $(y-y_0)\bmod c > c/2$;
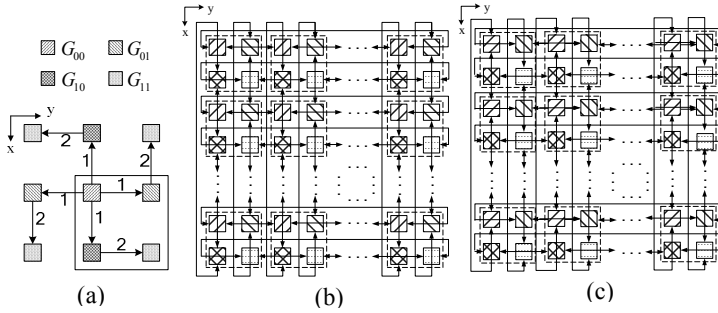QIV: if $(x-x_0)\bmod r \leq r/2$ and $(y-y_0)\bmod c > c/2$;

A. An Overview

The 2D torus can be regarded as the graph product of two rings, so we can apply the algorithm for the ring to construct our 2D algorithm. To fully utilize all the communication links, all nodes are divided into four groups, namely G_{00}, G_{01}, G_{10}, and G_{11}, according to the following rule:

$$G_{ij} = \{P(x,y)\mid x \bmod 2 = i \ \text{and} \ y \bmod 2 = j\}.$$

Hence, the original $r \times c$ torus is divided into $r/2 \times c/2$ sub-

Fig. 4 Communication pattern on all-port 2D torus

(a) Phase 1, (b) Phase 2, (c) Phase 3.

port		messages	destinations of messages
Lower	Port	$r \cdot c/8$	G_{10} and QI of $P(x_0, y_0)$
		$r \cdot c/16$	G_{11} and QI of $P(x_0, y_0)$
Right	Port	$r \cdot c/8$	G_{01} and QII of $P(x_0, y_0)$
		$r \cdot c/16$	G_{11} and QII of $P(x_0, y_0)$
Upper	Port	$r \cdot c/8$	G_{10} and QIII of $P(x_0, y_0)$
		$r \cdot c/16$	G_{11} and QIII of $P(x_0, y_0)$
Left	Port	$r \cdot c/8$	G_{01} and QIV of $P(x_0, y_0)$
		$r \cdot c/16$	G_{11} and QIV of $P(x_0, y_0)$

meshes of size 2×2. Each node in a 2×2 sub-mesh is included in one of four distinct groups (see Fig. 4(a)).

Firstly, we briefly describe a previous scheme proposed in [7, 8]. The scheme has two parts. In the first part, four nodes in each 2×2 sub-mesh exchange messages in two steps. After this part, each node in a 2×2 sub-mesh has messages originated from nodes in the same 2×2 sub-mesh and destined for nodes in the same group to which the node belongs. In the second part, nodes in the same group perform complete exchange among them to finish complete exchange. However, the previous scheme conforms to one-port constraint and fails to fully utilize all communication links; moreover, messages are not transmitted along shortest paths. The above problems hinder the previous scheme from achieving optimality in transmission time.

B. Communication Pattern

Inspired by our scheme on a ring, we design an ingenious scheme to solve the above problem. The proposed 2D algorithm consists of three phases. In Phase 1, each node in 2D torus sends messages destined for nodes in three other groups to its eight surrounding nodes, which respectively belong to three other groups (see Fig. 4(a)). Phase 1 requires two steps. In Step 1, every node utilizes its four ports to transmit messages to its four adjacent nodes simultaneously. For instance, we consider node $P(x_0, y_0) \in G_{00}$, the messages transmitted from node $P(x_0, y_0)$ are described in Table 1. Node $P(x_0, y_0)$ transmits $r \cdot c/8 + r \cdot c/16$ messages whose destination are in G_{10} (or G_{11}) and QI of $P(x_0, y_0)$ to its lower adjacent node $P(x_0+1, y_0)$, the same number of messages are transmitted to three other adjacent nodes, respectively and simultaneously. Our scheme can guarantee that all these messages are transmit- ted along shortest paths in the successive communication steps.

In Step 2, each node relays to transmit messages to its four adjacent nodes simultaneously. For instance, node $P(x_0+1, y_0)$ extracts these $r \cdot c/8$ messages meant for it and forwards remainder $r \cdot c/16$ messages to node $P(x_0+1, y_0+1)$.

After Phase 1, each node has received messages originated from its eight surrounding nodes and destined for $r \cdot c/4$ nodes in the same group to which the node belongs. Therefore, nodes in the same group perform complete exchange among them in Phase 2 and 3. If we consider a row, $c/2$ nodes in the group G_{00} (or G_{11}) can be regarded as a logical row ring. If we consider a column, $r/2$ nodes in the group G_{01} (or G_{10}) can also be regarded as a logical column ring. Hence, nodes in a row

TABLE I
COMMUNICATION PATTERN OF NODE $P(x_0, y_0)$

can transmit messages along two directions (row and column) in parallel without channel contention, so do nodes in a column.

In Phase 2, we can simultaneously run algorithm AR on every logical row ring and every logical column ring. In other words, nodes in the group G_{00} or G_{11} perform complete exchange along row direction, and nodes in the group G_{01} or G_{10} perform complete exchange along column direction simultaneously, (see fig. 4(b)). In Phase 3, nodes in the group G_{00} or G_{11} perform complete exchange along column direction, and nodes in the group G_{01} or G_{10} perform complete exchange along row direction, as showed in Fig. 4(c).

Note that there is no channel contention in our scheme, all communication links are fully utilized, and every message is transmitted along the shortest paths. The formal description of 2D algorithm for complete exchange in $r \times c$ all-port torus is shown in Fig. 5.

C. Data Array

Initially, we assume that each node $P(i, j)$ has $r \cdot c$ distinct messages to distribute other nodes in a 2D $r \times c$ torus, including one dummy message for itself. These messages are stored in a 2D array $A_{i,j}[0:r-1, 0:c-1]$, one message per location. If messages to be transmitted are not contiguous, then they should be rearranged before transmission. Before Phase 1, messages should be rearranged to facilitate our communication operations in Phase 1. After Phase 1, each node in the torus has received messages originated from its eight surrounding nodes. Regardless of the source nodes of the messages, each node has messages $M_{x,y \in A}^{(*,*)}$, where

$$A = \left\{ (x,y) \middle| \begin{array}{l} x = x_0, x_0 + c/2, x_0 \pm 2i, x_0 \pm 2i, 0 \le i \le r/4 - 1 \\ y = y_0, y_0 + r/2, y_0 \pm 2j, y_0 \pm 2j, 0 \le j \le c/4 - 1 \end{array} \right\}$$

Lemma 3. The communication cost in Phase 2 is:

$$T_{Phase2} = c/4 \cdot t_s + (r \cdot c/32 + r \cdot c/32 - r/8) \cdot c \cdot mt_w. \quad (3)$$

Proof. After Phase 1, each node has messages $M_{x,y \in A}^{(*,*)}$. In Phase 2, we can simultaneously run algorithm AR on every logical row ring and every logical column ring. Firstly, we consider a logical row ring of $c/2$ nodes. Each node $P(x_0, y_0)$ in logical row ring has c columns of messages destined for $c/2$ nodes. Except for one column of messages destined for $P(x_0, y_0 + c/2)$ and three columns destined for itself, node $P(x_0, y_0)$ has two columns of messages destined for every other node in the logical row ring. Thus, the communication cost for a logical row ring is:

Algorithm *AT2***:** //Complete Exchange on All-Port $r \times c$ Torus
BEGIN
  {Step 1 of Phase 1}
For each node $P(x_0, y_0)$ in the network Para_Do
$P(x_0, y_0)$ sends $M_{(x,y) \in R}^{(x_0, y_0)}$ to $P(x_0 + 1, y_0)$ (via Bottom-port)

$$B = \left\{ (x,y) \left| \begin{array}{ll} x = (x_0 + 2i - 1) \bmod r, & 1 \le i \le r/4 \\ y = (y_0 \bmod 2) \; or \; ((y_0 + 2j - 1) \bmod c), & 1 \le j \le c/4 \end{array} \right. \right\}$$

$P(x_0, y_0)$ sends $M_{(x,y) \in T}^{(x_0, y_0)}$ to $P(x_0, y_0 + 1)$ (via Right-port)

$$R = \left\{ (x,y) \left| \begin{array}{ll} x = (x_0 \bmod 2) \; or \; ((x_0 - 2i + 1) \bmod r), & 1 \le i \le r/4 \\ y = (y_0 + 2j - 1) \bmod c, & 1 \le j \le c/4 \end{array} \right. \right\}$$

$P(x_0, y_0)$ sends $M_{(x,y) \in L}^{(x_0, y_0)}$ to $P(x_0 - 1, y_0)$ (via Top-port)

$$T = \left\{ (x,y) \left| \begin{array}{ll} x = (x_0 - 2i + 1) \bmod r, & 1 \le i \le r/4 \\ y = (y_0 \bmod 2) \; or \; ((y_0 - 2j + 1) \bmod c), & 1 \le j \le c/4 \end{array} \right. \right\}$$

$P(x_0, y_0)$ sends $M_{(x,y) \in B}^{(x_0, y_0)}$ to $P(x_0, y_0 - 1)$ (via Left-port)

$$L = \left\{ (x,y) \left| \begin{array}{ll} x = (x_0 \bmod 2) \; or \; ((x_0 + 2i - 1) \bmod r), & 1 \le i \le r/4 \\ y = (y_0 - 2j + 1) \bmod c, & 1 \le j \le c/4 \end{array} \right. \right\}$$

  Endfor
  { Step 2 of Phase 1}
For each node $P(x_0, y_0)$ in the network Para_Do
  $P(x_0, y_0)$ sends $M_{(x,y) \in R^*}^{(x_0, y_0 + 1)}$ to $P(x_0 + 1, y_0)$ (via Bottom-port)

$$B^* = \left\{ (x,y) \left| \begin{array}{ll} x = (x_0 + 2i - 1) \bmod r, & 1 \le i \le r/4 \\ y = (y_0 - 2j + 2) \bmod c, & 1 \le j \le c/4 \end{array} \right. \right\}$$

  $P(x_0, y_0)$ sends $M_{(x,y) \in T^*}^{(x_0 - 1, y_0)}$ to $P(x_0, y_0 + 1)$ (via Right-port)

$$R^* = \left\{ (x,y) \left| \begin{array}{ll} x = (x_0 + 2i - 2) \bmod r, & 1 \le i \le r/4 \\ y = (y_0 + 2j - 1) \bmod c, & 1 \le j \le c/4 \end{array} \right. \right\}$$

  $P(x_0, y_0)$ sends $M_{(x,y) \in L^*}^{(x_0, y_0 - 1)}$ to $P(x_0 - 1, y_0)$ (via Top-port)

$$T^* = \left\{ (x,y) \left| \begin{array}{ll} x = (x_0 - 2i + 1) \bmod r, & 1 \le i \le r/4 \\ y = (y_0 + 2j - 2) \bmod c, & 1 \le j \le c/4 \end{array} \right. \right\}$$

  $P(x_0, y_0)$ sends $M_{(x,y) \in B}^{(x_0 + 1, y_0)}$ to $P(x_0, y_0 - 1)$ (via Left-port)

$$L^* = \left\{ (x,y) \left| \begin{array}{ll} x = (x_0 - 2i + 2) \bmod r, & 1 \le i \le r/4 \\ y = (y_0 - 2j + 1) \bmod c, & 1 \le j \le c/4 \end{array} \right. \right\}$$

  Endfor
  {Phase 2}
  (1) and (2) Para_Do
    (1) All nodes in G_{00} and G_{11} run algorithm *AR* to perform complete exchange along row direction;
    (2) All nodes in G_{01} and G_{10} run algorithm *AR* to perform complete exchange along column direction;
  {Phase 3}
  (1) and (2) Para_Do
    (1) All nodes in G_{00} and G_{11} run algorithm *AR* to perform complete exchange along column direction;
    (2) All nodes in G_{01} and G_{10} run algorithm *AR* to perform complete exchange along row direction;
  END.

Fig. 5 Description of Complete Exchange in All-Port Torus

$$T_{row\;ring} = c/4 \cdot t_s + (r \cdot c/32 + r \cdot c/32 - r/8) \cdot c \cdot mt_w$$

Secondly, we consider a logical column ring of $r/2$ nodes, and its communication cost is $T_{column\;ring}$. Since $r \le c$, $T_{row\;ring} \ge T_{column\;ring}$. Hence, the communication cost in Phase 2 is:

$$T_{Phase2} = c/4 \cdot t_s + (r \cdot c/32 + r \cdot c/32 - r/8) \cdot c \cdot mt_w. \qquad \square$$

Likewise, the communication cost in Phase 3 is:

$$T_{Phase3} = c/4 \cdot t_s + (r \cdot c/32 + r \cdot c/32 - r/8) \cdot c \cdot mt_w. \qquad (4)$$

Lemma 4. The total message transmission time of our complete exchange scheme on 2D $r \times c$ all-port torus, where r and c are multiples of four and $r \le c$, is $(r \cdot c^2/8) \cdot mt_w$, which completely achieves the theoretical lower bound.

Proof. Phase 1 consists of two steps and we should transmit $(r \cdot c/8 + r \cdot c/16)$ and $r \cdot c/16$ messages at each step, respectively. In Phase 2 and 3, there are $(r \cdot c/32 + r \cdot c/32 - r/8) \cdot c$ messages transmitted in each phase. So the total message transmission time is $(r \cdot c^2/8) \cdot mt_w$. $\qquad \square$

D. Complexity Analysis

We now analyze the time complexity of the proposed 2D algorithm in terms of startup time, message transmission time and rearrangement time.

1. *Startup time*: The 2D algorithm has three phases: Phase 1 has 2 steps, and $c/4$ steps per phase are required in Phase 2 and 3. Thus, the total startup time is $(c/2 + 2) \cdot t_s$.
2. *Message transmission time*: This is the time we spend in transmitting messages inside channels. According to lemma 4, the total message transmission time is $(r \cdot c^2/8) \cdot mt_w$, which completely achieves the theoretical lower bound.
3. *Rearrangement time*: This is the time we spend in rearranging messages between phases. At the beginning of each phase, messages are actually rearranged to prepare. Thus, the total data rearrangement time is $3 \cdot rc \cdot m\rho$.

V. PERFORMANCE ANALYSIS AND COMPARISON

In this section, the performance of the proposed algorithms is analyzed and compared with that of existing algorithms.

So far, we are not aware of any existing algorithms for complete exchange on all-port tori. Thus, we have to compare the performance of our algorithms with that of existing algorithms on one-port tori, regardless of port capability. For 2D tori, Tseng [5] and Suh [6] proposed indirect algorithms in which networks are assumed to be power-of-two and square tori. However, Suh's algorithm [10] and our algorithms can accommodate non-power-of-two tori where the number of nodes in each dimension needs not be power-of-two or square. Thus, Suh's algorithm [10] and our algorithm have better scalability.

TABLE II
COMPARISON OF COMPLETE EXCHANGE IN $2^d \times 2^d$ TORUS

	Startup Cost	Message Transmission Cost	Data Rearrangement Cost
[5]	$(2^{d-1} + 2) \cdot t_s$	$(2^{3d-2} + 2^{2d}) \cdot mt_w$	$(2^{d-1} + 1) \cdot 2^{2d} \cdot m\rho$
[6]	$(3d - 3) \cdot t_s$	$\{9 \cdot 2^{3d-4} + (d^2 - 5d + 3) \cdot 2^{2d-1}\} \cdot mt_w$	$\{9 \cdot 2^{3d-4} + (d^2 - 5d + 3) \cdot 2^{2d-1}\} \cdot m\rho$
[10]	$(2^{d-1} + 2) \cdot t_s$	$(2^{3d-2} + 2^{2d}) \cdot mt_w$	$3 \cdot 2^{2d} \cdot m\rho$
Proposed	$(2^{d-1} + 2) \cdot t_s$	$2^{3d-3} \cdot mt_w$	$3 \cdot 2^{2d} \cdot m\rho$

For comparison, the time complexities the proposed and other existing algorithms [5, 6, 10] on $2^d \times 2^d$ tori is presented in Table II. Obviously, the proposed algorithm is superior to Tseng [5] in terms of message transmission cost, and data rearrangement cost. Suh [6] achieves $O(d)$ startup cost, however, the constant associated with the transmission time is relatively high and the effect of this is significant as the message size is fairly large. In addition, the time complexity due to data

rearrangement is $O(2^{3d})$, while that of the other algorithms is $O(2^{2d})$. Though the startup time and data rearrangement time are equivalent to those in [10], the proposed algorithm completely achieves optimality in message transmission cost.

Because formal analysis of the scalability across a range of systems sizes is hampered by the lack of availability of a range of large system sizes, it is impossible to evaluate the performance of different algorithms on commercial parallel supercomputers. Therefore, the present studies are based on analytic models of execution time using values of parameters measured on the Intel Paragon: $t_s = 75\mu s$, $t_w = 0.011\mu s$, $\rho = 0.014\mu s$ [10].
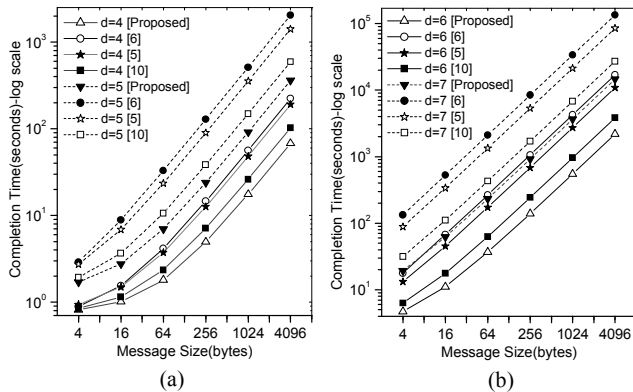

Fig. 6 Estimated performance of algorithms in 16×16, 32×32, 64×64, and 128×128 tori

Fig. 6 shows the expected completion time of the proposed 2D algorithm and existing algorithms [5, 6, 10] for various torus sizes as a function of message size. Obviously, the proposed algorithm always outperforms the other algorithms for any message size and any network size. Especially, message transmission cost becomes more important factor as network size and/or message size increase, so the proposed algorithm exhibits much better performance in large network and larger message size.

VI. CONCLUSIONS

In this paper, we have presented new indirect algorithms for complete exchange on all-port ring and 2D torus. These algorithms utilize message combining to reduce the startup time, take full advantage of all communication links, and send messages along shortest paths so as to completely achieve the theoretical lower bounds on transmission time. In addition, the proposed algorithms accommodate non-power-of-two tori where the number of nodes in each dimension needs not be power-of-two or square. Finally, the algorithms are conceptually simple and symmetrical for every message and every node so that they can be easily implemented.

The proposed algorithms can be used in tori with an arbitrary number of nodes in each dimension by adding imaginary nodes to compensate the network size. Our future research is to consider new complete exchange algorithms on multidimensional tori with an arbitrary number of ports.

REFERENCES

[1] P.K. McKinley, Y.J. Tsai, and D.F. Robinson, "A Survey of Collective Communication in Wormhole-Routed Massively Parallel Computers," Technical Report, MSU-CPS-94-35, Michigan State University, June 1994.
[2] S. Sur, H.W. Jin, and D.K. Panda, "Efficient and scalable all-to-all personalized exchange for infiniband-based clusters," *Proc. 2004 Int'l Conf. on Parallel Processing* (ICPP'04), pp. 275–282, Aug. 2004.
[3] C.C. Lam, C.H. Huang, and P. Sadayappan, "Optimal Algorithms for All-to-All Personalized Communication on Rings and Two Dimensional Tori," *Journal of Parallel and Distributed Computing*, No. 43, pp. 3–13, 1997.
[4] Y.C. Tseng and S.K.S. Gupta, "All-to-All Personalized Communication in a Wormhole-Routed Torus", *IEEE Trans. Parallel and Distributed Systems*, vol. 7, no. 5, pp. 498–505, May. 1996.
[5] Y.C. Tseng, T.H. Lin, S.K.S. Gupta and D.K. Panda, "Bandwidth-Optimal Complete Exchange on Wormhole- Routed 2D/3D torus Networks: A Diagonal-Propagation Approach," *IEEE Trans. Parallel and Distributed Systems*, vol. 8, no. 4, pp. 380-396, Apr. 1997.
[6] Y.J. Suh and S. Yalamanchili, "All-to-All Communication with Minimum Start-Up Costs in 2D/3D Tori and Meshes," *IEEE Trans. Parallel and Distributed Systems*, vol. 9, no. 5, pp. 442–458, May 1998.
[7] Y.C. Tseng, S.Y. Ni, and J.P. Sheu, "Toward Optimal Complete Exchange on Wormhole-Routed Tori," *IEEE Trans. Computers*, vol. 48, no. 10, pp. 1065-1082, Oct. 1999.
[8] GU Naijie, "Efficient Indirect All-to-All Personalized Communication on Rings and 2-D Tori," *Journal of Computer Science and Technology*, vol. 16, no. 5, pp. 480–483, Sept. 2001.
[9] N.S. Sundar, D.N. Jayasimha, D.K. Panda, and P. Sadayappan, "Hybrid Algorithms for Complete Exchange in 2D Meshes," *IEEE Trans. Parallel and Distributed Systems*, vol. 12, no. 12, pp. 1201–1218, Dec. 2001.
[10] Y.J. Suh and K.G. Shin, "All-to-All Personalized Communication in Multidimensional Torus and Mesh Networks," *IEEE Trans. Parallel and Distributed Systems*, vol. 12, no. 1, pp. 38–59, Jan. 2001.
[11] D.S. Scott, "Efficient All-to-All Communication Patterns in Hypercube and Mesh Topologies," *Proc. Sixth Conf. Distributed Memory Concurrent Computers*, pp. 398– 403, Portland, OR, April 1991.
[12] S.C. Chau and A.W.C. Fu, "An optical multistage interco- nnection network for optimal all-to-all personalized exchange," *Proc. fourth Int'l Conf. Parallel and Distributed Computing, Applications and Technologies* (PDCAT'2003), pp. 292–295, 27-29 Aug. 2003.
[13] Y.Y. Yang and J.C. Wang, "Near-Optimal All-to-All Broadcast in Multidimensional All-Port Meshes and Tori," *IEEE Trans. Parallel and Distributed Systems*, vol. 13, no. 2, pp. 128–141! Feb**.** 2002.
[14] J.P. Jung and I. Sakho, "A methodology for devising optimal all-port all-to-all broadcast algorithms in 2-dimensional tori,**"** 2*8th Annual IEEE Int'l Conf. Local Computer Networks* (LCN'03), pp. 558–566, Oct. 2003.

Liu Gang was born in Lanzhou, Gansu, China in 1978. He is a Ph.D. student in the Department of Computer Science and Technology, USTC. His research interests include interprocessor communication and mobile computing.
Gu Nai-jie was born in 1961. He is a Professor and Doctoral Advisor in the Department of Computer Science and Technology, USTC. His research interests include parallel computing architecture, interprocessor communication, and high-performance computing.
Bi Kun was born in 1981. He is a Ph.D. student in the Department of Computer Science and Technology, USTC. His research interests include parallel and distributed computing.
Tu Kun was born in 1980. He is a Ph.D. student in the Department of Computer Science and Technology, USTC. His research interests include parallel and distributed computing.
Dong Wan-li was born in 1981. He is a Ph.D. student in the Department of Computer Science and Technology, USTC. His research interests include parallel and distributed computing.