J Supercomput (2015) 71:894–908
DOI 10.1007/s11227-014-1342-3

The new Petersen-torus networks

Jong-Seok Kim · Hyeong-Ok Lee · Mihye Kim · Sung Won Kim

Published online: 25 November 2014

Abstract Routing and broadcasting are major parameters determining the performance of interconnection networks. In this paper, we propose a new Petersen-torus network NPT(m, n) by modifying the external edge definitions of the Petersen-torus network to improve its diameter and broadcasting times. We also show one-to-all broadcasting algorithms in NPT(m, n) using the single-link available and multiple-link available models.

Keywords Petersen-torus network · New Petersen-torus network · Routing · Diameter · Broadcasting

1 Introduction

A computer system is scalable if it can scale up its resources to accommodate demand for ever-increasing performance and functionality. In a parallel computer system with

J.-S. Kim
Department of Mathematics and Physics, North Carolina Central University, Durham, NC 27707, USA

H.-O. Lee
Department of Computer Education, Sunchon National University, Sunchon, Chonnam 540-742, South Korea

M. Kim
Department of Computer Science Education, Catholic University of Daegu, Gyeongsan, Gyeongbuk 712-702, South Korea

S. W. Kim (✉)
Department of Information and Communication Engineering, Yeungnam University, Gyeongsan, Gyeongbuk 712-749, South Korea
e-mail: swon@yu.ac.kr

a distributed-memory architecture, the design of the interconnection network topology is critical to the performance and scalability of the system. An interconnection network can be modeled as an undirected graph $G = (V, E)$, where $V(G)$ is the set of nodes and $E(G)$ is the set of edges of graph G. Each processor is an element of $V(G)$, and two arbitrary processors, u and v, are connected by a communication link (u, v). In G, each processor is represented as a node, and a communication link between two processors is represented as an edge. The distance between u and v in G is defined as the length of the shortest path connecting u and v, denoted $\mathrm{dist}(u, v)$. The diameter of G is defined as the maximal value of the distances between all pairs of nodes in G, denoted $\mathrm{diam}(G)$ (i.e., $\mathrm{diam}(G) = \max\{\mathrm{dist}(u, v) | u, v \in V(G)\}$).

Because a delay will occur whenever a packet passes through a node, the efficiency of communication can be improved by minimizing the diameter, and by minimizing the delay in transferring a packet from a source node to a destination node under the worst-case scenario for the network. As a result, with a given fixed number of interconnection resources (i.e., nodes and edges of an interconnection network), being able to construct an interconnection network with a diameter as small as possible is a very significant factor in the design of an interconnection network [1]. Broadcasting is also one of the major parameters determining the performance of interconnection networks, and is significantly influenced by the efficiency of broadcasting algorithms [2]. Broadcasting is a basic data communication method for interconnection networks, corresponding to message transmission between nodes [3]. In general, messages are disseminated between nodes in two ways: one-to-all broadcasting, whereby messages are sent from a source node to all other nodes in the network, and all-to-all broadcasting, where messages are sent from all nodes to all other nodes in the network [2–12]. Broadcasting algorithms are commonly based on two communication models: single-port or all-port communication [7,9]. In the single-port communication model, each node transmits messages using only one link incident on it at each stage of broadcasting, whereas in all-port communication, each node transmits messages using all links incident on it at each stage of broadcasting. The former is known as the single-link-available (SLA) model, and the latter is the multiple-link-available (MLA) model [13].

The mesh graph is one of the most well-known topologies for interconnection networks, and a number of variations of the mesh graph have been reported [14–21]. The Petersen-torus interconnection network $\mathrm{PT}(m, n)$ $(m, n \geq 2)$ is one such variation that was proposed by Seo et al. [22], and is based on the Petersen graph with a fixed four-degree network. The network costs are improved compared to mesh variation networks that have an equivalent number of nodes as $\mathrm{PT}(m, n)$. Properties including routing, broadcasting, and embedding were described, and advantages over mesh variation networks have been detailed [13,22–28]. The diameter of $\mathrm{PT}(m, n)$ has been shown to be $3(\mathrm{Max}(\lfloor \frac{m}{2} \rfloor, \lfloor \frac{n}{2} \rfloor)) + 2$ [22], and the one-to-all broadcasting time is $m + 2\log_2 m + 13$ with SLA model and $m + \log_2 m + 7$ with MLA model [13].

In this paper, we propose a new Petersen-torus network $\mathrm{NPT}(m, n)$ by modifying the external edge definitions specified in previous works [22,28] to improve its diameter and broadcasting times. We show that the diameter of $\mathrm{NPT}(m, n)$ is $2(\mathrm{Max}(\lfloor \frac{m}{2} \rfloor, \lfloor \frac{n}{2} \rfloor)) + 4$. In addition, we suggest algorithms for one-to-all and all-to-all broadcasting in $\mathrm{NPT}(m, n)$ using SLA and MLA models. The results show that one-to-all broadcasting of $\mathrm{NPT}(m, n)$ can be performed in $2\lfloor \frac{m}{2} \rfloor + 10$ with SLA model, and in $2\lfloor \frac{m}{2} \rfloor + 4$ with MLA model. And we show that the all-to-all broadcasting of $PT(m, n)$ can be performed in $2m + 2n + 8$ under SLA model. We also prove the all-to-all broadcasting time in $PT(m, n)$ under MLA model is $m + n + 5$ when m and n are equal to even and $2\lfloor \frac{m}{2} \rfloor + 2\lfloor \frac{n}{2} \rfloor + 6$ when m and n are equal to odd.

The remainder of this paper is organized as follows. In Sect. 2, we describe the properties of $PT(m, n)$ and propose a new Petersen-torus network $NPT(m, n)$. In Sect. 3, we suggest a simple routing algorithm and the diameter of $NPT(m, n)$. In Sect. 4, we describe one-to-all broadcasting algorithms for $NPT(m, n)$ under SLA and MLA models. In Sect. 5, we describe all-to-all broadcasting algorithms for $NPT(m, n)$ under SLA and MLA models. Section 6 summarizes and concludes the paper.

2 Petersen-torus network $PT(m, n)$ and new Petersen-torus network $NPT(m, n)$

The Petersen-torus network $PT(m, n)$ $(m, n \geq 2)$ is based on the Petersen graph [29], which is a regular node- and edge-symmetric graph with 10 nodes, a degree of 3, and a diameter of 2. $PT(m, n)$ is also a regular graph, and has $10mn$ nodes, $20mn$ edges, and a fixed degree of 4. $PT(m, n)$ is defined as follows: $PT(m, n) = (V_{pt}, E_{pt})$, where V_{pt} is a set of nodes and $E_{pt}$is a set of edges. An edge that connects two arbitrary nodes A and B is denoted (A, B). A node in $PT(m, n)$ is represented by Definition 1 [22].

Definition 1 $V_{pt} = \{(x, y, p), 0 \leq x \leq m, 0 \leq y \leq n, 0 \leq p \leq 9\}$.

In $PT(m, n)$, a Petersen graph is located at the intersection of the X- and Y-axes on a coordinate plane, and is called as a module. The address of a module is represented by (x, y) and the address of a node in a module by (x, y, p), where x and y are the X- and Y-axes of the module and p is a node address in the module (i.e., in the Petersen graph). The edges can be divided into internal and external edges, where an internal edge connects two arbitrary nodes in a module (i.e., an internal edge is that of the Petersen graph), and an external edge connects two nodes located in different modules. Definition 2 describes the external edges of $PT(m, n)$ [22], where the symbol % represents the modulus operator.

Definition 2 1) The longitudinal edges are $((x, y, 6), (x, (y + 1)\%n, 9))$.
2) The latitudinal edges are $((x, y, 1), ((x + 1)\%m, y, 4))$.
3) The diagonal edges are $((x, y, 2), ((x + 1)\%m, (y + 1)\%n, 3))$.
4) The reverse-diagonal edges are $((x, y, 7), ((x - 1 + m)\%m, (y + 1)\%n, 8))$.
5) The diameter edges are $((x, y, 0), ((x + \lfloor \frac{m}{2} \rfloor)\%m, (y + \lfloor \frac{n}{2} \rfloor)\%n, 5))$.
6) The wraparound edge is $((x, 0, 9), (x, n-1, 6))$ $((0, y, 4), (m-1, y, 1)$ $((0, y, 7), (m - 1, (y + 1)\%n, 8))$ $((0, y, 3), (m - 1, (y - 1 + n)\%n, 2))$ $((x, 0, 8), ((x + 1)\%m, n - 1, 7))$ $((x, 0, 3), ((x - 1 + m)\%m, n - 1, 2))$.

Now, we propose a new Petersen-torus network $NPT(m, n)$ by replacing the external edge definitions in Definition 2 with those given in Definition 3.

Definition 3 1) The longitudinal edges are $((x, y, 6), (x, (y + 1)\%n, 2))$.
2) The latitudinal edges are $((x, y, 8), ((x + 1)\%m, y, 1))$.
3) The diagonal edges are $((x, y, 9), ((x + 1)\%m, (y + 1)\%n, 3))$.
4) The reverse-diagonal edges are $((x, y, 7), ((x - 1 + m)\%m, (y + 1)\%n, 4))$.

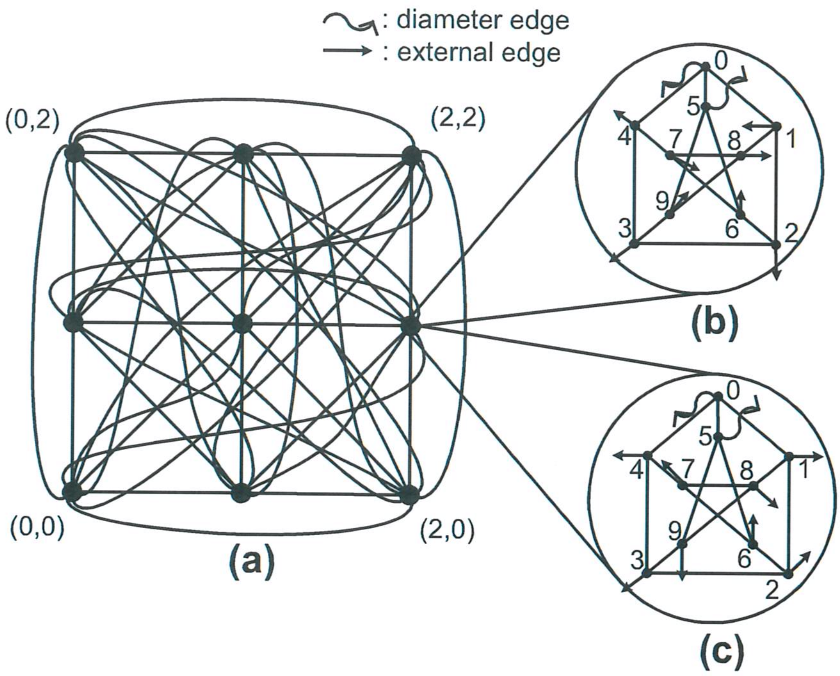

Fig. 1 **a** A new Peterson-torus network NPT(3, 3), **b** a Petersen graph in NPT(3, 3), and **c** a Petersen graph in PT(3, 3)

5) The diameter edges are $((x, y, 5), ((x + \lfloor \frac{m}{2} \rfloor)\%m, (y + \lfloor \frac{n}{2} \rfloor)\%n, 0))$.
6) The wraparound edges are $((x, 0, 2), (x, n-1, 6))$ $((0, y, 1), (m-1, y, 8)$ $((0, y, 4), (m-1, (y+1)\%n, 7))$ $((0, y, 3), (m-1, (y-1+n)\%n, 9))$ $((x, 0, 7), ((x+1)\%m, n-1, 4))$ $((x, 0, 3), ((x-1+m)\%m, n-1, 9))$.

Figure 1a shows a new Petersen-torus network NPT(3, 3) and Fig. 1b shows a Petersen graph in NPT(3, 3); both were constructed using the external edges defined in Definition 3. Figure 1c shows a Petersen graph in PT(3, 3) constructed using the external edges defined in Definition 2.

3 Routing algorithm and diameter of NPT(m, n)

In this section, we propose a simple routing algorithm for NPT(m, n) $(m, n \geq 2)$. Let two arbitrary nodes of NPT(m, n) be $v_s = (x_s, y_s, z_s)$ and $v_d = (x_d, y_d, z_d)$, and let the module that includes node v_s be $M_s = (x_s, y_s)$ and the module that includes node v_d be $M_d = (x_d, y_d)$. Assume that module M_s is a source module and M_d is a destination module. If $M_s = M_d$, two nodes are inside the same module because the addresses of the two modules are the same. Therefore, the maximum distance between the two nodes is 2. In the routing algorithms reported here, we only describe routing by external edges, because routing by internal edges is equivalent to that described in the previous study [22]. We denote $|x_s - x_d|$ as d_x (i.e., $d_x = |x_s - x_d|$) and $y_s - y_d$ as d_y (i.e., $d_y = |y_s - y_d|$). We also divide the routing area into four regions according to the positions of M_s and M_d. If $d_x \leq \lfloor \frac{m}{2} \rfloor$ and $d_y \leq \lfloor \frac{n}{2} \rfloor$, module M_d is in area A, and if $d_x > \lfloor \frac{m}{2} \rfloor$ and $d_y > \lfloor \frac{n}{2} \rfloor$, module M_d is in area B. If $d_x \leq \lfloor \frac{m}{2} \rfloor$ and $d_y > \lfloor \frac{n}{2} \rfloor$, module M_d is in area C, and if $d_x > \lfloor \frac{m}{2} \rfloor$ and $d_y \leq \lfloor \frac{n}{2} \rfloor$, module M_d is in area D. Here, we

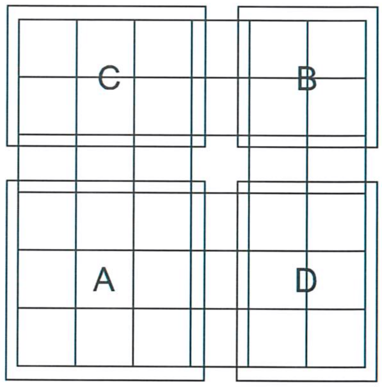

Fig. 2 An example of the four routing areas in NPT(7, 7)

describe the advanced routing algorithm only for area A, because routing for the other areas can be simulated in a way similar to that for area A. Figure 2 shows the routing area divided into four areas within NPT(7, 7), which is depicted using longitudinal and latitudinal edges.

In NPT(m, n), the routing distance inside a module, which connects two external edges (except for diameter edges), depends on the types of external edges used. The routing distance inside a module via two longitudinal (or two latitudinal) edges is 1, and via a longitudinal (or latitudinal) and a latitudinal (or longitudinal) edge is either 2 or 1. The routing distance inside a module via two diagonal (or two reverse-diagonal) edges is 1, and via a diagonal (or reverse-diagonal) and a latitudinal (or longitudinal) edge is either 2 or 1. Let a module that connects diagonal and longitudinal (or latitudinal) edges be $M_c = (x_c, y_c)$. When the destination module M_d belongs to area A, the simple routing algorithm (SRA) is as follows:

1) If $d_x = d_y$ or $d_x = 0$ or $d_y = 0$, then $M_s \Longrightarrow M_d$.
2) Otherwise, $M_s \Longrightarrow M_c \Longrightarrow M_d$.

In case 1), when $d_x = d_y$, the routing is processed from M_s to M_d via only diagonal edges; when $d_x = 0$, the routing is processed via only longitudinal edges; and when $d_y = 0$, the routing is processed via only latitudinal edges. In case 2), the routing from module M_s to module M_d is processed from M_s to M_c via diagonal edges and from M_c to M_d via longitudinal (or latitudinal) edges. In this case, there are eight routing paths $P_1, P_2, \ldots, P_8$, as shown in Fig. 3.

The condition and M_c for each path are as follows:

$P_1 : d_x < d_y$ and $x_d - x_s > 0$ and $y_d - y_s > 0$, $M_c = (x_c = x_d, y_c = y_s + d_x)$
$P_2 : d_x > d_y$ and $x_d - x_s > 0$ and $y_d - y_s > 0$, $M_c = (x_c = x_s + d_y, y_c = y_d)$
$P_3 : d_x > d_y$ and $x_d - x_s > 0$ and $y_d - y_s < 0$, $M_c = (x_c = x_s + d_y, y_c = y_d)$
$P_4 : d_x < d_y$ and $x_d - x_s > 0$ and $y_d - y_s < 0$, $M_c = (x_c = x_d, y_c = y_s - d_x)$
$P_5 : d_x < d_y$ and $x_d - x_s < 0$ and $y_d - y_s < 0$, $M_c = (x_c = x_d, y_c = y_s - d_x)$
$P_6 : d_x > d_y$ and $x_d - x_s < 0$ and $y_d - y_s < 0$, $M_c = (x_c = x_s - d_y, y_c = y_d)$
$P_7 : d_x > d_y$ and $x_d - x_s < 0$ and $y_d - y_s > 0$, $M_c = (x_c = x_s - d_y, y_c = y_d)$
$P_8 : d_x < d_y$ and $x_d - x_s < 0$ and $y_d - y_s > 0$, $M_c = (x_c = x_d, y_c = y_s + d_x)$

Let us assume a module that is connected to module M_s via a diameter edge is $M_t = (x_t, y_t)$. There exist two M_t modules in NPT(m, n) where m or n is odd. Let these two modules be M_{t_1} and M_{t_2}. To obtain the minimum routing distance between

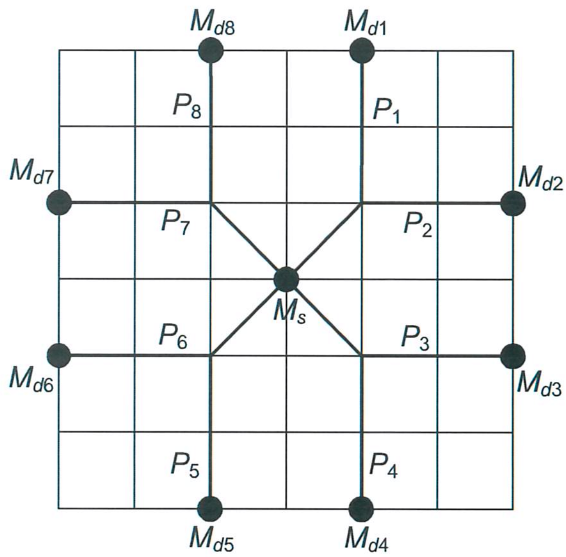

Fig. 3 An example of eight routing paths in area A of NPT(m, n)

modules M_s and M_d, routing is required via M_t. Unless the following conditions are met, routing between M_s and M_d is performed using SRA in NPT(m, n).

Condition 1. When both m and n are even: 1-1) z_s is 0 and $d_x + d_y \geq |x_t - x_d| + |y_t - y_d| + 1$. 1-2) $\text{dist}((x_s, y_s, z_s), (x_s, y_s, 0))$ is 1 and $d_x + d_y \geq |x_t - x_d| + |y_t - y_d| + 2$. 1-3) $\text{dist}((x_s, y_s, z_s), (x_s, y_s, 0))$ is 2 and $d_x + d_y \geq |x_t - x_d| + |y_t - y_d| + 3$.

Condition 2. When m or n is odd and $M_t = M_{t_1}$: 2-1) z_s is 0 and $d_x + d_y \geq |x_t - x_d| + |y_t - y_d| + 1$. 2-2) $\text{dist}((x_s, y_s, z_s), (x_s, y_s, 0))$ is 1 and $d_x + d_y \geq |x_t - x_d| + |y_t - y_d| + 2$. 2-3) $\text{dist}((x_s, y_s, z_s), (x_s, y_s, 0))$ is 2 and $d_x + d_y \geq |x_t - x_d| + |y_t - y_d| + 3$.

Condition 3. When m or n is odd and $M_t = M_{t_2}$: 3-1) z_s is 5 and $d_x + d_y \geq |x_t - x_d| + |y_t - y_d| + 1$. 3-2) $\text{dist}((x_s, y_s, z_s), (x_s, y_s, 5))$ is 1 and $d_x + d_y \geq |x_t - x_d| + |y_t - y_d| + 2$. 3-3) $\text{dist}((x_s, y_s, z_s), (x_s, y_s, 5))$ is 2 and $d_x + d_y \geq |x_t - x_d| + |y_t - y_d| + 3$.

If one of the above conditions is met, routing between M_s and M_d is as follows:

1) Perform routing from M_s to M_t using diameter edges.
2) Perform routing from M_t to M_d using the simple routing algorithm in NPT(m, n).

diam(NPT(m, n)) via the simple routing algorithm is the total of: (the routing distance inside module M_s) + (the routing distance inside module M_c) + (the routing distance inside module M_d) (the routing distances inside modules in SRA except for modules M_s, M_c, and M_d) + (the number of external edges that are used in SRA). Therefore, diam(NPT(m, n)) is $2(\text{Max}(\lfloor \frac{m}{2} \rfloor, \lfloor \frac{n}{2} \rfloor)) + 4 = 2 + 2 + 2 + \text{Max}(\lfloor \frac{m}{2} \rfloor, \lfloor \frac{n}{2} \rfloor) - 2 + \text{Max}(\lfloor \frac{m}{2} \rfloor, \lfloor \frac{n}{2} \rfloor)$.

Theorem 1 diam(NPT(m, n)) *is* $2(\text{Max}(\lfloor \frac{m}{2} \rfloor, \lfloor \frac{n}{2} \rfloor)) + 4$.

Note. Based on Definition 2, Seo et al. showed that diam$(PT(m, n))$ is $3(\text{Max}(\lfloor \frac{m}{2} \rfloor, \lfloor \frac{n}{2} \rfloor)) + 2$ [22], and reduced it to $2(\text{Max}(\lfloor \frac{m}{2} \rfloor, \lfloor \frac{n}{2} \rfloor)) + 4$ using an optimal routing algorithm in a previous study [28]. The proof proceeds as follows: "In the intermediate module between successive latitudinal edges, the length of internal path is 2, and in the intermediate module between successive diagonal edges, it is

1" [28]. "At worst, routing may be done only by diagonal edges, so the internal path length for intermediate module is $2(\mathrm{Max}(\lfloor\frac{m}{2}\rfloor, \lfloor\frac{n}{2}\rfloor)) - 1$" [28]. Here, the worst case is "only via latitudinal edges", and is not "only by diagonal edges". Therefore, the diameter in that study must be $3(\mathrm{Max}(\lfloor\frac{m}{2}\rfloor, \lfloor\frac{n}{2}\rfloor)) + 2$, which follows from the sum of $2(\mathrm{Max}(\lfloor\frac{m}{2}\rfloor, \lfloor\frac{n}{2}\rfloor)) - 1$ (from the internal path lengths of intermediate modules), $\mathrm{Max}(\lfloor\frac{m}{2}\rfloor, \lfloor\frac{n}{2}\rfloor)$ (from the external path lengths of intermediate modules), and 4 (the internal routing from a source module to a destination module), even though it is based on optimal routing.

For example, let two arbitrary nodes in $\mathrm{PT}(8, 8)$ be $u = (0, 0, 9)$ and $v = (4, 0, 2)$. Here, a diameter edge does not exist between module (0,0), to which node u belongs, and module (0,4), to which node v belongs. Therefore, routing from node u to node v must be as follows:
$(0, 0, 9) \rightarrow (0, 0, 8) \rightarrow (0, 0, 1) \rightarrow (1, 0, 4) \rightarrow (1, 0, 0) \rightarrow (1, 0, 1) \rightarrow (2, 0, 4) \rightarrow (2, 0, 0) \rightarrow (2, 0, 1) \rightarrow (3, 0, 4) \rightarrow (3, 0, 0) \rightarrow (3, 0, 1) \rightarrow (4, 0, 4) \rightarrow (4, 0, 3) \rightarrow (4, 0, 2)$. From this routing, we can see that the distance from node u to node v is 14; that is, $\mathrm{diam}(\mathrm{PT}(m, n))$ is $3(\mathrm{Max}(\lfloor\frac{m}{2}\rfloor, \lfloor\frac{n}{2}\rfloor)) + 2$, rather than $2(\mathrm{Max}(\lfloor\frac{m}{2}\rfloor, \lfloor\frac{n}{2}\rfloor)) + 4$, from Definition 2.

4 One-to-all broadcasting NPT(m, n)

Since the diameter of $\mathrm{NPT}(m, n)$ is $O(m + n)$, any broadcasting algorithm, under SLA or MLA model, has a lower bound of $\Omega(m + n)$. The broadcasting of $\mathrm{NPT}(m, n)$ must be calculated by dividing the network into two cases: $m \geq n$ and $n \geq m$. However, we only analyze the case of $m \geq n$, because the results of the two cases are the same. The one-to-all broadcasting of the Petersen graph was analyzed as Lemma 1.

Lemma 1 [13] *The one-to-all broadcasting time of the Petersen graph is 4 in SLA model and 2 in MLA model.*

In the one-to-all broadcasting algorithm (OBA), if $x = 0$, then $x - 1 = m - 1$, if $x = m - 1$, then $x + 1 = 0$, if $y = 0$, then $y - 1 = n - 1$, and if $y = n - 1$, then $y + 1 = 0$ where $0 \leq x \leq m - 1, 0 \leq y \leq n - 1$. The following symbols are defined for OBA.

- M_0: The basic module to which the source node belongs.
- M_s: Modules that receive a message through diagonal or reverse-diagonal edges and forward the message to adjacent modules via longitudinal or latitudinal edges.
- $v_i = (x, y, z_1)$; an internal node of an arbitrary module where $0 \leq z_1 \leq 9, 0 \leq i \leq 9$.
- w_j: An internal node of a module adjacent to the module to which v_i belongs.
- $\longrightarrow$: Message transmission via an internal edge.
- $\hookrightarrow$: Message transmission via an external edge.

The conditions for one-to-all broadcasting for $\mathrm{NPT}(m, n)$ under SLA and MLA models are as follows:

Conditions for SLA Model:

Condition 1. All nodes without a message located inside M_0 receive the message from the source node and perform Step 1 in OBA.

Condition 2. The modules that receive a message through a reverse-diagonal edge transmit the message in the following order, and the modules that receive a message through diagonal, longitudinal, and latitudinal edges also transmit the message in a way similar to the following order. However, at 5), if the modules to receive a message (i.e., the modules to which nodes w_1 and w_2 belong) already have the message, the message is not sent to the modules.

1) $v_0 = (x, y, 7) \longrightarrow v_1 = (x, y, 4)$.
2) $v_1 = (x, y, 4) \hookrightarrow w_0 = (x-1, y+1, 7)$; $v_0 = (x, y, 7) \longrightarrow v_2 = (x, y, 8)$.
3) $v_0 = (x, y, 7) \longrightarrow v_3 = (x, y, 6)$; $v_1 = (x, y, 4) \longrightarrow v_4 = (x, y, 3)$; $v_2 = (x, y, 8) \longrightarrow v_5 = (x, y, 1)$.
4) $v_1 = (x, y, 4) \longrightarrow v_6 = (x, y, 0)$; $v_2 = (x, y, 8) \longrightarrow v_7 = (x, y, 9)$; $v_3 = (x, y, 6) \longrightarrow v_8 = (x, y, 5)$; $v_4 = (x, y, 3) \longrightarrow v_9 = (x, y, 2)$.
5) $v_5 = (x, y, 1) \hookrightarrow w_1 = (x-1, y, 8)$; $v_3 = (x, y, 6) \hookrightarrow w_2 = (x, y+1, 2)$.

Condition 3. The modules transmit a message in a way similar to Condition 1 when all their eight adjacent modules (except for the modules connected via diameter edges) already have the message.

Conditions for MLA Model:

Condition 1. All nodes without a message located inside M_0 receive the message from the source node and perform Step 1 in OBA.

Condition 2. The modules that receive a message through a reverse-diagonal edge transmit the message in the following order, and the modules that receive a message through diagonal, longitudinal, and latitudinal edges also transmit the message in a way similar to the following order. However, at 3), if the modules to receive a message already have the message, the message is not sent to the modules.

1) $v_0 = (x, y, 7) \longrightarrow v_1 = (x, y, 4)$; $v_0 = (x, y, 7) \longrightarrow v_2 = (x, y, 8)$; $v_0 = (x, y, 7) \longrightarrow v_3 = (x, y, 6)$.
2) $v_1 = (x, y, 4) \hookrightarrow w_0 = (x-1, y+1, 7)$; $v_1 = (x, y, 4) \longrightarrow v_4 = (x, y, 0)$; $v_1 = (x, y, 4) \longrightarrow v_5 = (x, y, 3)$; $v_2 = (x, y, 8) \longrightarrow v_6 = (x, y, 1)$; $v_2 = (x, y, 8) \longrightarrow v_7 = (x, y, 9)$; $v_3 = (x, y, 6) \longrightarrow v_8 = (x, y, 2)$; $v_3 = (x, y, 6) \longrightarrow v_9 = (x, y, 5)$.

3) $v_6 = (x, y, 1) \hookrightarrow w_1 = (x-1, y, 8)$; $v_3 = (x, y, 6) \hookrightarrow w_2 = (x, y+1, 2)$.

Condition 3. The modules transmit a message in a way similar to Condition 1 when all their eight adjacent modules (except for the modules connected via diameter edges) already have the message.

Table 1 shows OBA of NPT(m, n) under SLA and MLA models.

Figure 4 illustrates one-to-all broadcasting of NPT(16, 16) with SLA model. Numbers represent the arrival time of a message to the corresponding module, and numbers in parentheses represent the arrival time of a message to all nodes in the module. Arrows represent the processes for message transmission.

Table 1 One-to-all broadcasting algorithm (OBA) for SLA and MLA models of NPT(m, n)

Step 1. A message is transmitted from a source module with the message to eight modules adjacent to the source module through an external edge
Step 2. The modules that receive a message via diagonal or reverse-diagonal edges follow procedures under Condition 2
Step 3. Repeat Step 2 until the message is transmitted to all modules in NPT(m, n)

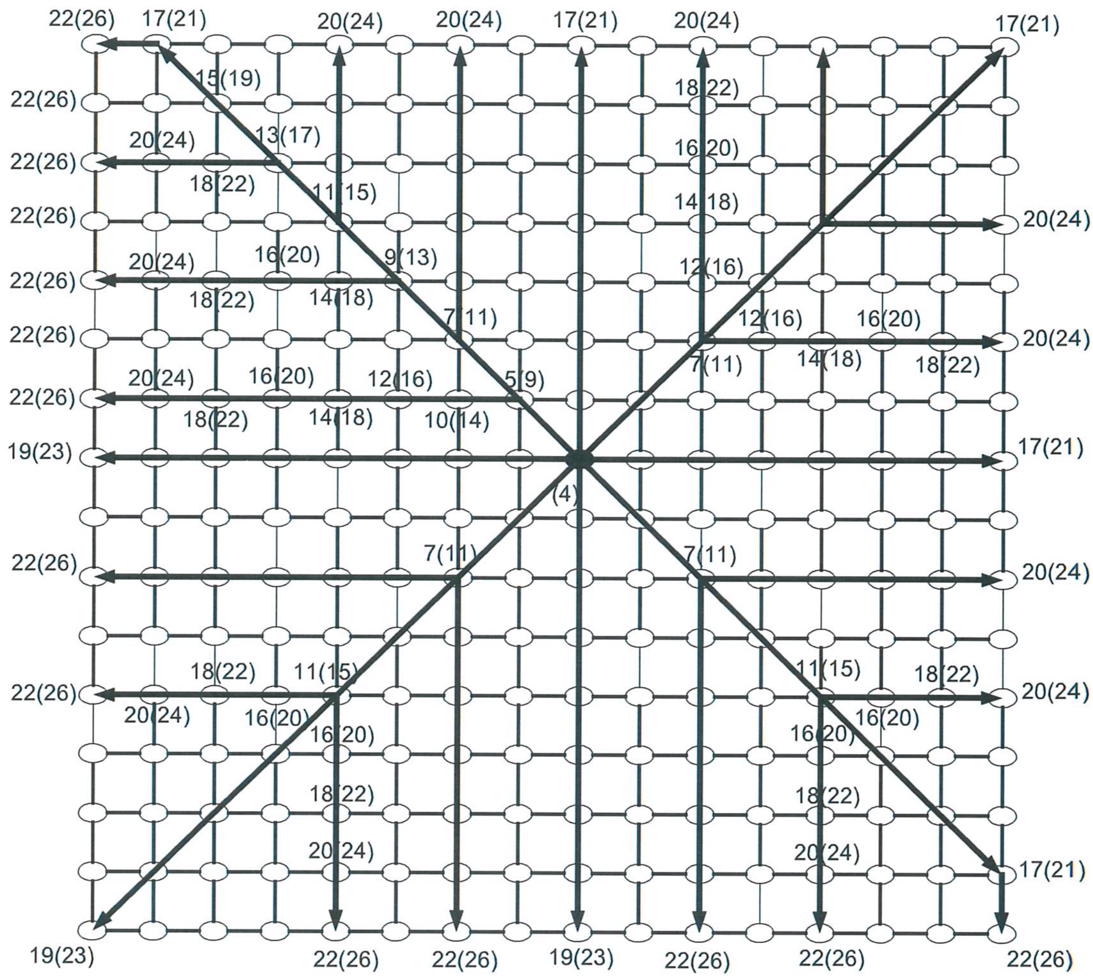

Fig. 4 Example of one-to-all broadcasting of NPT(16, 16) under SLA model

Theorem 2 *The one-to-all broadcasting times of* NPT(m, n) *are* $2\lfloor \frac{m}{2} \rfloor + 10$ *and* $2\lfloor \frac{m}{2} \rfloor + 4$ *when broadcasting is based on SLA model and MLA model, respectively.*

Proof Theorem 2 is proved by dividing the broadcasting time into two cases depending on the number of edge types used for broadcasting.

Case 1. Broadcasting is performed via only one type of external edge, such as diagonal, reverse-diagonal, longitudinal, or latitudinal edge:

The internal broadcasting time for the source module is 4 with SLA model and 2 with MLA model by Lemma 1. When broadcasting is based on SLA model or MLA model, the maximum broadcasting time via an external edge is $\lfloor \frac{m}{2} \rfloor$. With SLA or

Table 2 Comparison of the diameter and broadcasting times obtained with PT(n, n) and NPT(n, n) for the same dimension n

n	Diameter		Broadcasting times under SLA model		Broadcasting times under MLA model	
	PT(n, n)	NPT(n, n)	PT(n, n)	NPT(n, n)	PT(n, n)	NPT(n, n)
4	8	8	17	14	13	8
8	14	12	27	18	18	12
16	26	20	37	26	27	20
32	50	36	55	42	47	36
64	98	68	89	74	77	68
128	194	132	155	138	142	132

MLA models, the maximum total value from performing $(x, y, 7) \longrightarrow (x, y, 4)$ in each module in NPT(m, n) except for the source and destination modules is $\lfloor \frac{m}{2} \rfloor - 1$. The internal broadcasting time for the destination module is 4 with SLA model and 2 with MLA model by Lemma 1. Therefore, the broadcasting time for this case is $4 + \lfloor \frac{m}{2} \rfloor + \lfloor \frac{m}{2} \rfloor - 1 + 4 = 2\lfloor \frac{m}{2} \rfloor + 7$ and $2 + \lfloor \frac{m}{2} \rfloor + \lfloor \frac{m}{2} \rfloor - 1 + 2 = 2\lfloor \frac{m}{2} \rfloor + 3$ when using SLA and MLA models, respectively.

Case 2. Broadcasting is performed via the combination of two types of external edge, such as reverse-diagonal and latitudinal edges, or reverse-diagonal and longitudinal edges:

We assume that the two edges used are reverse-diagonal and longitudinal edges.The internal broadcasting time for the source module is 4 with SLA model and 2 with MLA model by Lemma 1. When broadcasting is based on SLA model or MLA model, the maximum broadcasting time via an external edge is $\lfloor \frac{m}{2} \rfloor$. The internal broadcasting time of module M_s is 4 with SLA model and 2 with MLA model, because the interior of M_s, which receives a message via a reverse-diagonal edge and transmits the message to adjacent modules through a longitudinal edge, must adhere to Condition 2. With SLA model or MLA model, the maximum total value from performing $(x, y, 7) \longrightarrow (x, y, 4)$ or $(x, y, 8) \longrightarrow (x, y, 1)$ in each module in NPT(m, n), except for the source, destination, and M_s modules, is $\lfloor \frac{m}{2} \rfloor - 2$. The internal broadcasting time for the destination module is 4 with SLA model and 2 with MLA model by Lemma 1. Thus, the broadcasting time in this case is $4 + \lfloor \frac{m}{2} \rfloor + 4 + \lfloor \frac{m}{2} \rfloor - 2 + 4 = 2\lfloor \frac{m}{2} \rfloor + 10$ when broadcasting is based on SLA model, and $2 + \lfloor \frac{m}{2} \rfloor + 2 + \lfloor \frac{m}{2} \rfloor - 2 + 2 = 2\lfloor \frac{m}{2} \rfloor + 4$ when broadcasting is based on MLA model. □

Table 2 lists a comparison of the diameter and broadcasting times under SLA model and under MLA model between PT(n, n) and NPT(n, n). Figure 5 shows the results obtained with graphs for the same dimension n.

5 All-to-all broadcasting NPT(m, n)

Lemma 2 *All-to-all broadcasting time of a Petersen graph using SLA model is 6, and the broadcasting time of a Petersen graph using MLA model is 3.*

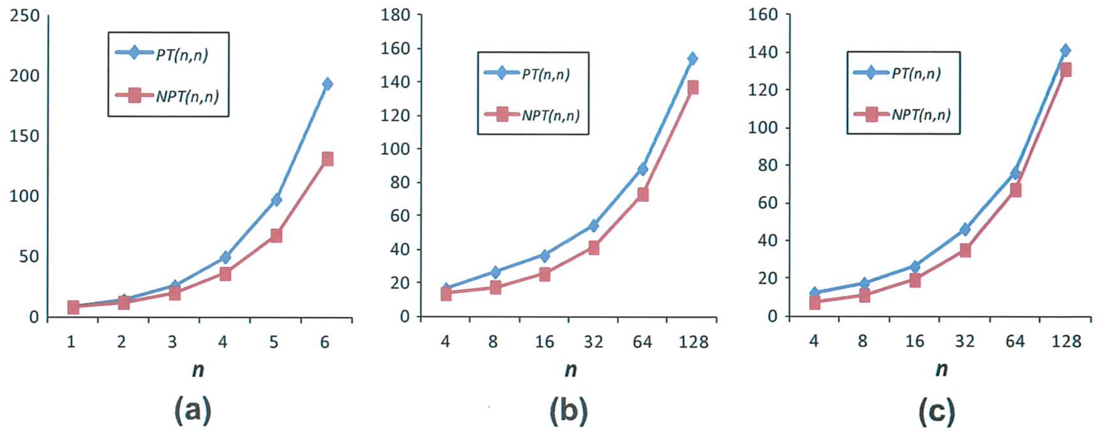

Fig. 5 **a** The diameter, **b** the broadcasting times under SLA model, and **c** the broadcasting times under MLA model of PT(n, n) and NPT(n, n) for the same dimension n

Proof We can see that there are two cycles, which are represented as a by $\{0,1,2,3,4\}$ and b by $\{5,6,7,8,9\}$ in the Petersen graph. A broadcasting mechanism using SLA model is as follows. In parallel, both 5-cycles a and b can do an all-to-all broadcasting in four time units. Now, each node on cycle a has five messages from all five nodes on the cycle, and the same can be said for nodes on cycle b. In another step, all nodes on cycle b send their five messages to their corresponding nodes on cycle a. In one more step, all nodes on cycle a send their five messages to their corresponding nodes on cycle b so that now, all nodes in the Petersen graph have ten messages, and the total time is 6.

The broadcasting mechanism using MLA model is as follows. In the first step of broadcasting, all the nodes on a and b cycles send messages to all the adjacent nodes. After this step is repeated once, all the nodes on a and b cycles will have messages from all the nodes on each cycle. In the second step, all the nodes on cycle a and all the nodes on cycle b that are linked via edges will send messages to, and get messages from, each other. Therefore, all-to-all broadcasting time in a Petersen graph using MLA model is 3. □

Table 3 shows the all-to-all broadcasting algorithm of NPT(m, n) using SLA ($0 \leq x \leq m-1, 0 \leq y \leq n-1$). In this algorithm, if $x = 0$ then $x - 1 = m - 1$, and if $y = 0$ then $y - 1 = n - 1$.

All-to-all broadcasting time using SLA is as follows. Step 1 is a process that sends messages of a node within each module to all the other nodes within the same basic module that constitutes NPT(m, n) and its broadcasting time is 6. Step 2 is a process that sends messages of $(x, y, 8)$ that are nodes of the basic module located in all the columns within NPT(m, n), and its broadcasting time is $2m - 3$. Step 3 is a process that sends messages $(x, y, 1) - (x, y, 2) - (x, y, 6)$, and its broadcasting time is 2. Step 4 is a process that sends messages of $(x, y, 6)$ that are located in all the rows within NPT(m, n), and its broadcasting time is $2n - 3$. Step 5 is a process that sends the messages of $(x, y, 2)$ within each basic module to all the other nodes within the same basic module, and its broadcasting time is 6. Therefore, all-to-all broadcasting time using SLA is $2m + 2n + 8$.

Table 3 All-to-all broadcasting algorithm (SABA) of NPT(m, n) using SLA

Step 1. Send messages of all nodes within each module to all the other nodes within the same basic module that constitutes NPT(m, n)

Step 2. Using internal edges and vertical edges, send messages to all the basic modules on all the columns in NPT(m, n): $(x, y, 8) - ((x+1)\%m, y, 1) - ((x+1)\%m, y, 8) - ((x+2)\%m, y, 1) - ((x+2)\%m, y, 8) - \cdots - ((x-1+m)\%m, y, 1)$

Step 3. Send messages $(x, y, 1) - (x, y, 2) - (x, y, 6)$

Step 4. Using internal edges and horizontal edges, send messages to all the basic modules on all the rows in NPT(m, n): $(x, y, 6) - (x, (y+1)\%n, 2) - (x, (y+1)\%n, 6) - (x, (y+2)\%n, 2) - (x, (y+2)\%n, 6) - \cdots - (x, (y-1+n)\%n, 2)$

Step 5. Send messages of $(x, y, 2)$ within each basic module to all the other nodes within the same basic module that constitutes NPT(m, n)

Theorem 3 *Broadcasting time of* NPT(m, n) *from the all-to-all broadcasting algorithm using SLA is* $2m + 2n + 8$.

Conditions for all-to-all broadcasting of NPT(m, n) **under MLA model:**
Condition 1. Using internal edges and horizontal edges, send messages to all the basic modules on all the columns in NPT(m, n):

1) $m =$ even: $(x, y, 8) - ((x+1)\%m, y, 1) - ((x+1)\%m, y, 8) - ((x+2)\%m, y, 1) - ((x+2)\%m, y, 8) - \cdots - ((x+\frac{m}{2})\%m, y, 1)$ and $(x, y, 1) - ((x-1+m)\%m, y, 8) - ((x-1+m)\%m, y, 1) - ((x-2+m)\%m, y, 8) - ((x-2+m)\%m, y, 1) - \cdots - ((x+\frac{m}{2})\%m, y, 8)$.
2) $m =$ odd: $(x, y, 8) - ((x+1)\%m, y, 1) - ((x+1)\%m, y, 8) - ((x+2)\%m, y, 1) - ((x+2)\%m, y, 8) - \cdots - ((x+\lfloor\frac{m}{2}\rfloor)\%m, y, 1)$

and $(x, y, 1) - ((x-1+m)\%m, y, 8) - ((x-1+m)\%m, y, 1) - ((x-2+m)\%m, y, 8) - ((x-2+m)\%m, y, 1) - \cdots - ((x+\lceil\frac{m}{2}\rceil)\%m, y, 8)$.

Condition 2. Using internal edges and vertical edges, send messages to all the basic modules on all the rows in NPT(m, n):

1) $n =$ even: $(x, y, 6) - (x, (y+1)\%n, 2) - (x, (y+1)\%n, 6) - (x, (y+2)\%n, 2) - (x, (y+2)\%n, 6) - \cdots - (x, (y+\frac{n}{2})\%n, 2)$ and $(x, y, 2) - (x, (y-1+n)\%n, 6) - (x, (x, (y-1+n)\%n, 2) - (x, (x, (y-2+n)\%n, 6) - (x, (x, (y-2+n)\%n, 2) - \cdots - (x, (y+\frac{n}{2})\%n, 6)$.
2) $n =$ odd: $(x, y, 6) - (x, (y+1)\%n, 2) - (x, (y+1)\%n, 6) - (x, (y+2)\%n, 2) - (x, (y+2)\%n, 6) - \cdots - (x, (y+\lfloor\frac{n}{2}\rfloor)\%n, 2)$

and $(x, y, 2) - (x, (y-1+n)\%n, 6) - (x, (x, (y-1+n)\%n, 2) - x, (x, (y-2+n)\%n, 6) - ((x, (x, (y-2+n)\%n, 2) - \cdots - (x, (y+\lceil\frac{n}{2}\rceil)\%n, 6)$.

Table 4 shows the all-to-all broadcasting algorithm (MABA) of NPT(m, n) using MLA $(0 \le x \le m-1, 0 \le y \le n-1)$. In this algorithm, if $x = 0$ then $x - 1 = m - 1$, and if $y = 0$ then $y - 1 = n - 1$.

All-to-all broadcasting time using MLA is as follows. Step 1 is a process that sends message of a node within each module to all the other nodes within the same basic module that constitutes NPT(m, n), and its broadcasting time is 3. Step 2 is a process

Table 4 All-to-all broadcasting algorithm of NPT(m, n) using MLA

Step 1. Send messages of all nodes within each module to all the other nodes within the same basic module that constitutes NPT(m, n)
Step 2. Perform Condition 1 under conditions for all-to-all broadcasting of NPT(m, n) under MLA model
Step 3. Send messages $(x, y, 1) - (x, y, 2) - (x, y, 6)$ when $m =$ even, $(x, y, 1) - (x, y, 2) - (x, y, 6)$ and $(x, y, 8) - (x, y, 1) - (x, y, 2)$ and $(x, y, 8) - (x, y, 5) - (x, y, 6)$ when $m =$ odd
Step 4. Perform Condition 2 under conditions for all-to-all broadcasting of NPT(m, n) under MLA model
Step 5. Send messages of $(x, y, 2)$ and $(x, y, 6)$ within each basic module to all the other nodes within the same basic module that constitutes NPT(m, n)

that sends messages of $(x, y, 1)$ and $(x, y, 8)$ that are nodes of the basic module located in all the columns within NPT(m, n), and its broadcasting time is $m-1$ when m is equal to even and $2\lfloor \frac{m}{2} \rfloor - 1$ when m is equal to odd. Step 3 is a process that sends messages $(x, y, 1)-(x, y, 2)-(x, y, 6)$ when m is equal to even, $(x, y, 1)-(x, y, 2)-(x, y, 6)$ and $(x, y, 8) - (x, y, 1) - (x, y, 2)$ and $(x, y, 8) - (x, y, 5) - (x, y, 6)$ when m is equal to odd, and its broadcasting time is 2. Step 4 is a process that sends messages of $(x, y, 2)$ and $(x, y, 6)$ that are located in all the rows within NPT(m, n), and its broadcasting time is $n - 1$ when n is equal to even and $2\lfloor \frac{n}{2} \rfloor - 1$ when n is equal to odd. Step 5 is a process that sends the messages of $(x, y, 2)$ and $(x, y, 6)$ within each basic module to all the other nodes within the same basic module, and its broadcasting time is 2 when n is equal to even and 3 when n is equal to odd. Therefore, all-to-all broadcasting time using MLA is $m + n + 5$ when m and n are equal to even and $2\lfloor \frac{m}{2} \rfloor + 2\lfloor \frac{n}{2} \rfloor + 6$ when m and n are equal to odd.

Theorem 4 *Broadcasting time of* NPT(m, n) *from the all-to-all broadcasting algorithm using MLA is* $m + n + 5$ *when* m *and* n *are equal to even and* $2\lfloor \frac{m}{2} \rfloor + 2\lfloor \frac{n}{2} \rfloor + 6$ *when* m *and* n *are equal to odd.*

In view of the $\Omega(m + n)$ lower bound, all of our broadcasting algorithms are asymptotically optimal.

6 Conclusion

Routing, diameter, and broadcasting are major parameters determining the performance of interconnection networks. In this paper, we proposed a new Petersen-torus network NPT(m, n) by modifying the external edge definitions of the previous PT(m, n), and we showed that the diameter of NPT(m, n) is $2(\mathrm{Max}(\lfloor \frac{m}{2} \rfloor, \lfloor \frac{n}{2} \rfloor)) + 4$ using the simple routing algorithm. We also proposed the one-to-all broadcasting algorithm for NPT(m, n) using both SLA and MLA models, resulting in one-to-all broadcasting times of $2\lfloor \frac{m}{2} \rfloor + 10$ and $2\lfloor \frac{m}{2} \rfloor + 4$, respectively. And we showed that the all-to-all broadcasting of PT(m, n) can be performed in $2m + 2n + 8$ under SLA model. In addition, we proved that the all-to-all broadcasting time in PT(m, n) under MLA model is $m + n + 5$ when m and n are equal to even and $2\lfloor \frac{m}{2} \rfloor + 2\lfloor \frac{n}{2} \rfloor + 6$ when m and n are equal to odd. Therefore, the routing and broadcasting methods reported here are expected to be extremely useful for analysis of the properties of NPT(m, n), including optimal routing, parallel routing algorithm, and embedding.

Acknowledgments We thank the reviewers for their comments and suggestions which have substantially improved our presentation. This work was supported by the 2012 Yeungnam University Research Grant. Also, this research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology(2012R1A1A4A01014439).

References

1. Zhou W, Fan J, Jia X, Zhang S (2011) The spined cube: a new hypercube variant with smaller diameter. Inf Process Lett 111:561–567
2. Johnsson SL (1987) Communication efficient basic linear algebra computations on hypercube architectures. J Parallel Distrib Comput 16:133–172
3. Mendia VE, Sarkar D (1992) Optimal broadcasting on the star graph. IEEE Trans Parallel Distrib Syst 3(4):389–396
4. Albader B, Bose B, Flahive M (2012) Efficient communication algorithms in hexagonal mesh interconnection networks. IEEE Trans Parallel Distrib Syst 23(1):69–77
5. Bai L, Maeda H, Ebara H, Nakano H (1998) A broadcasting algorithm with time and message optimum on arrangement graphs. J Graph Algorithms Appl 2(2):1–17
6. Carle J, Myoupo J-F, Seme D (1999) All-to-all broadcasting algorithms on honeycomb networks and applications. Parallel Process Lett 9(4):539–550
7. Deazevedo MM, Bagherzadeh N, Latifi S (1995) Broadcasting algorithms for the star-connected cycles interconnection network. J Parallel Distrib Comput 25:209–222
8. Farah RN, Othman M (2014) Broadcasting communication in high degree modified chordal rings networks. Appl Math Inf Sci 8(1):229–233
9. Mkwawa IM, Kouvatsos DD (2003) An optimal neighborhood broadcasting scheme for star interconnection networks. J Interconnect Netw 4(1):103–112
10. Stewart IA (2014) Interconnection networks of degree three obtained by pruning two-dimensional tori. IEEE Trans Comput 63(10):2473–2486
11. Thomson A, Zhou S (2014) Frobenius circulant graphs of valency six, Eisenstein–Jacobi networks, and hexagonal meshes. Eur J Comb 38:61–78
12. Yang X, Wang L, Yang L (2012) Optimal broadcasting for locally twisted cubes. Inf Process Lett 112:129–134
13. Seo JH, Lee HO (2009) One-to-all broadcasting in Petersen-torus networks for SLA and MLA model. ETRI J 31(3):327–329
14. Dally W, Seitz C (1986) The torus routing chip. Distrib Comput 1:187–196
15. Chen MS, Shin KG (1990) Addressing, routing, and broadcasting in hexagonal mesh multiprocessors. IEEE Trans Comput 39(1):10–18
16. Tang KW, Padubidri SA (1994) Diagonal and toroidal mesh networks. IEEE Trans Comput 43(7):815–826
17. Stojmenovic I (1997) Honeycomb network: topological properties and communication algorithms. IEEE Trans Parallel Distrib Syst 8(10):1036–1042
18. Arabnia HR, Oliver MA (1989) A transputer network for fast operations on digitised images. Int J Eurograph Assoc (Computer Graphics Forum) 8(1):3–12
19. Arabnia HR (1990) A parallel algorithm for the arbitrary rotation of digitized tmages using process-and-data-decomposition approach. J Parallel Distrib Comput 10(2):188–193
20. Bhandarkar SM, Arabnia HR (1995) The REFINE multiprocessor: theoretical properties and algorithms. Parallel Comput 21(11):1783–1806
21. Arabnia HR, Smith JW (1993) A reconfigurable interconnection network for imaging operations and its implementation using a multi-stage switching box. In: Proceedings of the 7th annual international high performance computing conference. The 1993 high performance computing: new horizons supercomputing symposium, pp 349–357

22. Seo JH, Lee HO, Jang M (2008) Petersen-torus networks for multicomputer systems. In: Proceedings of the international conference on networked computing and advanced information management, pp 567–571
23. Seo JH, Sim H, Park DH, Park JW, Lee YS (2011) One-to-one embedding between honeycomb mesh and Petersen-torus networks. Sensors 11:1959–1971
24. Seo JH, Jang M, Kim E, Ban K, Ryu N, Lee HO (2009) One-to-one embedding between hyper Petersen and Petersen-torus networks. Commun Comput Inf Sci 63:133–139
25. Seo JH, Lee HO, Jang M, Kim E (2008) Node mapping algorithm between hypercube and Petersen-torus networks. In: Proceedings of the international conference on networked computing and advanced information management, pp 535–539
26. Seo JH, Lee HO, Jang M, Han SH (2008) Node mapping algorithm between torus and Petersen-torus networks. In: Proceedings of the international conference on networked computing and advanced information management, pp 540–544
27. Seo JH, Lee HO, Jang M (2008) Constructing complete binary trees on Petersen-torus networks. In: Proceedings of the international conference on computer sciences and convergence information technology, pp 252–255
28. Seo JH, Lee HO, Jang M (2008) Optimal routing and hamiltonian cycle in Petersen-torus networks. In: Proceedings of the international conference on computer sciences and convergence information technology, pp 303–308
29. Holton DA, Sheehan J (1993) The Petersen graph. Australian Mathematical Society lecture notes, vol 7. Cambridge University Press, London