Research Article

Link-Disjoint Broadcasting Algorithm in Wormhole-Routed 3D Petersen-Torus Networks

Jung-hyun Seo¹ and HyeongOk Lee²

¹ Department of Computer Engineering, National University of Suncheon Chonnam Suncheon 540-950, Republic of Korea
² Department of Computer Education, National University of Suncheon, Jungang-ro 255, Chonnam, Suncheon 540-950, Republic of Korea

Correspondence should be addressed to HyeongOk Lee; oklee@scnu.ac.kr

Received 22 March 2013; Revised 15 November 2013; Accepted 17 November 2013

Academic Editor: Yanmin Zhu

In a network, one-to-all broadcasting is the process of disseminating messages from a source node to all the nodes existing in the network through successive data transmissions between pairs of nodes. Broadcasting is the most primary communication process in a network. A 3D Petersen-Torus (3D PT) network has been proposed recently: the three-dimensionally magnified Petersen-Torus topology, 3D PT, is suitable for establishing a wireless sensor network (WSN) in 3D rectangular or cylindrical structures such as buildings. We propose a link-disjoint broadcasting algorithm for half-duplex 3D PT networks with wormhole routing and prove that the broadcasting algorithm is link-disjoint and the broadcasting step is $l/2 + \log_2(n-1) + 5$ in 3D PT ($l, n, n$).

1. Introduction

In wireless sensor network (WSN), after sensor nodes are deployed, they self-organize and establish routes automatically and transmit their information on their surroundings to a base station (BS). Since each sensor node has a limited and irreplaceable energy resource, energy conservation is the most important performance consideration in a WSN [1]. To decrease the energy consumption of a node, it is very important to effectively transmit collected data to a router through a sink node. For effective message transmission, a routing path should be short, without message congestion. When the routing path is long, the number of nodes used for message transmission increases, leading to additional energy consumption. Further, link congestion results in excessive energy consumption and lower transfer rate.

A routing protocol influenced by node topology is crucial to reducing energy consumption. Topologies used for effective routing include star [2], peer-to-peer (P2P) [3], grid mesh [4], and hexagonal honeycomb [5]. The Petersen-Torus, a topology that connects a Petersen graph in the grid shape, involves lower network costs than both the grid mesh and the honeycomb. 3D PT, the three-dimensionally magnified Petersen-Torus topology, is suitable for establishing a wireless sensor network (WSN) in 3D rectangular or cylindrical structures such as buildings. 3D PT involves lower network costs than the 3D mesh or the 3D honeycomb [6]. This study proposes broadcasting for data transmission in a WSN topology using 3D PT. As the proposed broadcasting is edge-disjoint, it does not cause message congestion. The time complexity, which is $O(n)$ at 3D PT ($n, n, n$), is confirmed to be efficient.

It is anticipated that sequential computers will reach their physical limitations in terms of CPU speed and memory size. Over the past years, high performance computing (HPC) has been extensively studied by researchers and applied to a variety of fields ranging from science, engineering, and business. With the development of HPC, there are many computing paradigms including multicomputer, multiprocessor, grid computing, cloud computing, and peer-to-peer (P2P) computing [7]. The interconnection network (structure of interconnecting processors) plays a critical role in improving the system performance of a HPC and wireless sensor network (WSN). A great deal of research on HPC and WSN has dealt with interconnection networks to achieve goals such as low latency, high bandwidth, low hardware cost, and low power consumption. The performance of HPC and WSN can be improved with advancements in hardware
implementation technology, routing schemes, broadcasting algorithm, data/task distribution, and many other application parameters [6, 8].

In a network, processors (nodes) often need to communicate with each other for various reasons, such as data migration, message collection, job allocation, and message dissemination. Efficient communication has been recognized to be critical for HPC. Broadcasting is one of the most fundamental tasks in network communication. One essential communication operator is one-to-all broadcast. A broadcasting algorithm is applicable in diversified manners in several fields such as management of shared variables for distributed programming, image processing, data copying in database of large-scale network, and data collection in WSN, and for this, an effective broadcasting algorithm is necessary [9].

In a HPC, according to composition of processor and communication link, interconnection network can be classified into mesh family, hypercube family, and star graph family. The 3D Petersen-Torus network is mesh class and it was announced recently. An interconnection network can be modeled as an undirected graph \( G = (V, E) \), where the node set \( V \) represents the processors and the edge set \( E \) represents the communication links. Each processor \( P_i \) is an element of node set \( V_i \) and two processors \( P_i \) and \( P_j \) are connected by communication links \( (P_i, P_j) \). The number of nodes adjacent to the node \( P_i \) is defined as degree of the node.

Broadcasting is performed by successive calling to transmit a message and has several limitations [10]: first, just two nodes participate in a call; second, a call is fully done in unit time; third, each node may take part in one call in unit time; and, fourth, a call appears between the neighboring two nodes. The third condition depends upon whether or not routing equipment supports one-port communication or all-port communication. The one-port communication means that one node holding a message in unit time transmits the message just to the one neighboring node, while the all-port communication means that one node holding a message in unit time transmits the message to all the neighboring nodes. In this paper, we assume a communication model where each communication channel is half duplex and each node has all-port capability. While store-and-forward routing was used in the first generation networks, most new generation multicomputers use wormhole-like routing [11]. Machines adopting such technology include the Cray T3D [12], MIT’s J-Machine [13], and Intel Touchstone DELTA [14]. Such networks are known to be insensitive to the routing distance if the communication path is contention free. This study is on condition of distance-insensitivity wormhole routing that distance between nodes does not influence communication latency. Therefore the fourth condition is neglected, and it is important not to cause link contention in broadcasting.

The broadcasting algorithm of 3D PT\((k, n, n)\) proposed in this study, is divided into two stages. A basic module refers to a bundle of 10 nodes that take the form of a Petersen graph in 3D PT. In the first stage, a message is transmitted from the source node that possessed it to the \( k \) basic module(s) in a different 2D space. In the second stage, the \( k \) node(s) possessing the message transmits it to every basic module located in the same 2D space. In this stage, the divide-and-conquer method, which continuously divides 2D basic modules into four areas, is used. As the divided areas do not overlap, the transmitted message does not cause congestion. Once eight basic modules exist in the divided area, not including the targeted basic module, the division process terminates. When the division process stops, the message is transmitted to the eight basic modules that take the form of a complete graph. In the final stage, every basic module that includes one node with a message transmits the message from the nodes including that message to the other nodes.

For broadcasting, constructing the spanning tree in a network is frequently used. Using the binary tree is suitable for either hypercube class or star graph class network, where the more the number of nodes is, the more the degree is. Using the binomial tree is also available in a more improved way. Time complexity of a one-to-all broadcasting algorithm of the all-port model in a 3D mesh class network is as follows: a broadcasting algorithm of\([15]\) in \(2^n \times 2^n \times 2 \) torus is \( k + 1 \), and that of\([16]\) in \((a + 1)^k \times (a + 1)^k \times (a + 6) \) torus is \( 3k + 1 \). A broadcasting algorithm of\([17]\) in \( k \times k \times k \) torus is \( 3[\log_k k] + 2 \). Besides, there are several algorithms for broadcasting in mesh class\([18–20]\). Section 2 introduces the 3D Petersen-Torus network and analyzes a broadcasting algorithm in the Petersen graph. Section 3 proposes a one-to-all broadcasting algorithm and proves that broadcasting is edge-disjoint. And, finally, conclusion is made.

2. Relevant Studies

2.1. 3D PT\((l, m, n)\) Network. Consider 3D PT\((l, m, n) = (VPT, EPT)\). The Petersen graph is set as a basic module, and the address of the basic module is indicated by \((z, x, y)\), while the node address is indicated by \((z, x, y, p)\). It is assumed that \((z, x, y, 0), (z, x, y, 1), \ldots, (z, x, y, 9)\) belong to the basic module \((z, x, y)\). \( z \) is a coordinate of the \( z \)-axis of the basic module, \( x \) is a coordinate of the \( x \)-axis of the basic module, \( y \) is a coordinate of the \( y \)-axis of the basic module, and \( p \) is the address of the node in the Petersen graph of the basic module. The node of 3D PT\((l, m, n)\) is defined as

\[
VPT = \{(z, x, y, p), 0 \leq z < l, 0 \leq x < m, 0 \leq y < n, 0 \leq p \leq 9\}.
\]

The edge of 3D PT\((l, m, n)\) is divided into an internal edge and an external edge, as mentioned below. Edges connecting the nodes belonging to the same basic module are called internal edges, and, for internal edges, those of the Petersen graph are used as they are. Edges connecting nodes in different basic modules are called external edges and are defined below [6]. In the equations expressing the following edges, the symbol “/” is a modulation operator.

(1) The vertical edge is \(((z, x, y, 6), (z, x, (y + 1)/n, 9))\).

(2) The horizontal edge is \(((z, x, y, 1), (z, x + 1/m, y, 4))\).

(3) The diagonal edge is \(((z, x, y, 2), (z, x + 1/m, (y + 1)/n, 3))\).
Figure 1: 3D Petersen-Torus network.

(a) 3D PT(5,5,5)

(b) Basic module

Figure 2: Broadcasting step in the Petersen graph.

(a) One-to-all-broadcasting

(b) Message transfer from node 2 to nodes 3, 7, and 8

2.2. Edge-Disjoint Broadcast on Petersen Graph. This section examines the broadcast of the Petersen graph, which is the basis of the broadcast of the 3D PT. Petersen graph is a regular graph, a node- and edge-symmetry graph, degree 3, diameter 2, connectivity 3, and girth 5 [21].

Lemma 1. The one-to-all broadcasting in the Petersen graph is edge-disjoint and broadcasting time complexity is 2.

Proof. The process of receiving a message on condition of 2 of the broadcasting step in all nodes is as shown in Figure 2(a), with node 0 in the Petersen graph as a start node. Figure 2(a) shows the Petersen graph in a form of a spanning tree. A message is transmitted from root node to child nodes and the message is transmitted from the three child nodes to each terminal node as shown in Figure 2(a); therefore the broadcasting step is 2. The Petersen graph is a node- (edge-) symmetry graph; therefore the broadcasting step from a certain node to all nodes of the Petersen graph is 2. As shown in Figure 2(a), the paths are edge-disjoint with each other in the ① and ③ steps, and the demonstration is omitted.

Lemma 2. In the Petersen graph, three broadcasting paths to transmit a message from node 2 to nodes 3, 7, and 8 are edge-disjoint.

Proof. Figure 2(b) shows the Petersen graph in a form of a spanning tree. As shown in Figure 2(b), the three paths indicated in a solid line are edge-disjoint with each other, and the demonstration is omitted.

3. Edge-Disjoint Broadcast on 3D PT Network

In this section, a broadcasting algorithm is proposed. The time complexity (hereafter referred to as broadcasting time)
of the suggested algorithm is calculated, and it is demonstrated that the algorithm is edge-disjoint. \( U(z, x, y, p) \) is assumed to be a source node with an initial message. The broadcasting process is divided into three stages. First, a message is transferred from the source node to \( l \) nodes with the address of \((z', x, y, 2)\) (here, \(0 \leq z' < l\)). Next, a message from \( l \) nodes with an address of \((z', x', y', 2)\) is transferred to \( l(mn-1) \) nodes with an address of \((z, x', y', 2)\) (here, \(0 \leq x' < m, 0 \leq y' < n\)). Once this transfer is completed, the node, where \( p = 2 \) in all the basic modules in a network, has a message. Finally, as shown in Figure 2(a), all nodes with a message transfer it to all the other nodes present in the basic modules to which they belong. The broadcasting process is shown in Figure 3. In the next section, the symbol "%" signifies the modulation operator and "→" signifies the path created by the routing algorithm that is presented in [6].

**Stage 1.** Consider

\[
U(z, x, y, p) \rightarrow (z, x, y, 2) \rightarrow (z, x, y, 2)
\]

\[
\rightarrow ((z + 1) \mod l, x, y, 2)
\]

\[
\rightarrow ((z + 2) \mod l, x, y, 2)
\]

\[
\rightarrow ((z + 3) \mod l, x, y, 2)
\]

\[
\rightarrow \cdots \rightarrow \left(\left(z + \frac{l}{2}\right) \mod l, x, y, 2\right)
\]

\[
\rightarrow (z, x, y, 2) \rightarrow ((z - 1 + l) \mod l, x, y, 2)
\]

\[
\rightarrow ((z - 2 + l) \mod l, x, y, 2)
\]

\[
\rightarrow \cdots \rightarrow \left(\left(z - \frac{l}{2} + l\right) \mod l, x, y, 2\right).
\]

Processes (a) and (b) proceed simultaneously and get completed at time \([l/2]\). Consequently, the broadcasting time of Stage 1 is \([l/2] + 1\). Once this stage is completed, only one node has a message in all the subgraphs (the two-dimensional maps in Figure 3) with the same value of \(z\). The number of nodes with a message is \(l\).

**Stage 2.** In Stage 2, \(l\) nodes simultaneously transfer the message to one node in all the other basic modules, each with a different value of \(z\). When comparing two cases, where \(z = 0\) and \(0 < z < l - 1\), it is observed that the results of both cases are the same. Seeing just a case of \(z = 0\), transfer process is as follows. The source node is \(U(0, x, y, 2)\), while the destination node is \(V(0, x', y', 2)\). Take \(dx = \left|x - x'\right|\) and \(dy = \left|y - y'\right|\). \(d = dx\) is an \(x\)-axis distance of the basic module to which node \(U\) and node \(V\) belong, while \(dy\) is a \(y\)-axis distance. Broadcasting uses a divide-and-conquer technique. With a basic module having the initial message as an axis, it is divided into 4 areas, and the message is transmitted to each basic module located in the center of the 4 areas. With 4 basic modules having a message as an axis, each area is divided again into four subareas, and the message is transmitted to basic modules located in the center of each area. The said division and message transmission are repeated until \(dx(dy) < 1\). Once division is finished, a message is transmitted from a basic module having a message to the 8 neighboring basic modules. This process is as shown in Figure 4. The numbers in the circle indicate the order that the message is transmitted from one basic module to another basic module. A subgraph on condition of \(z = 0\), where the basic module is indicated in dots, is as shown in Figure 4, a node- (edge-) symmetry graph. Therefore nodes with messages are randomly selected. It is assumed that \(m = n\). Consider

\[
U(0, x, y, 2) \rightarrow \left(0, x + \left(\frac{m}{4}\right), y + \left(\frac{n}{4}\right), 2\right)
\]

\[
\rightarrow \left(0, x + \left(\frac{m}{8}\right), y + \left(\frac{n}{8}\right), 2\right)
\]

\[
\rightarrow \left(0, x + \left(\frac{m}{16}\right), y + \left(\frac{n}{16}\right), 2\right)
\]

\[
\rightarrow \cdots
\]
Processes (c), (d), (e), and (f) proceed simultaneously. A message is transmitted from $U(0, x, y, 2)$ to 4 basic modules on condition of $dx(dy) = [n/4]$. In the next step, $dx$ is a half of $dx$ of the previous step, $dx(dy) = [n/8]$. To meet $dx(dy) < 1$, the division time should be done in $\lceil \log_2(n-1) \rceil - 1$. When $dx(dy) < 1$, division should be stopped and a message is transmitted to the 8 neighboring basic modules. To transmit a message to the neighboring basic modules, the message should be transmitted to all nodes of the basic modules to which the node having the message belongs. Finally, transmit a message to the neighboring basic modules. Once this process is completed, only one node has a message in all the basic modules of 3D PT.

Stage 3. A message is transferred from the node $(z, x, y, 2)$ in all the basic modules to all the other nodes in the basic module to which it belongs, as shown in Figure 2(a). Broadcasting occurs along an edge-disjoint path and the broadcasting time is 2.

Theorem 3. On 3D PT($l, n, n$), the broadcasting step is $l/2 + \log_2(n-1) + 5$ and the broadcasting algorithm is edge-disjoint.

Proof. In Stage 1, broadcasting step is $\lceil l/2 \rceil + 1$. In Stage 2, initially, $dx(dy) = [n/4]$, and in the next step, $dx$ is a half of $dx$ of the previous step, $dx(dy) = [n/8]$. To meet $dx(dy) < 1$, the division time should be done in $\lceil \log_2(n-1) \rceil - 1$. When $dx(dy) < 1$, division should be stopped and a message is transmitted to the 8 neighboring basic modules. To transmit a message to the neighboring basic modules, the message should be transmitted to all nodes of the basic modules to which the node having the message belongs. As described in Lemma 1, the step is 2 and the broadcasting step to transmit a message to the neighboring basic modules is 1. In Stage 3, broadcasting step is 2 according to Lemma 1. Adding up the broadcasting step $T = \lceil l/2 \rceil + 1 + \lceil \log_2(n-1) \rceil - 1 + 2 + 1 + 2$. In the worst case, $T = l/2 + \log_2(n-1) + 5$. In Stage 1, all nodes existing on the paths except for subpath $(z, x, y, 2) \rightarrow ((z + 1)/l, x, y, 2)$ for (a) and subpath $(z, x, y, 2) \rightarrow ((z - 1 + l)/l, x, y, 2)$ for (b) are edge-disjoint with different addresses on the $z$-axis. When subpath $(z, x, y, 2) \rightarrow ((z + 1)/l, x, y, 2)$ for (a) is unfolded, $(z, x, y, 2) \rightarrow (z, x, y, 1) \rightarrow (z, x, y, 0) \rightarrow ((z + 1)/l, x, y, 5) \rightarrow ((z + 1)/l, x, y, 6) \rightarrow ((z + 1)/l, x, y, 2)$, and when subpath $(z, x, y, 2) \rightarrow ((z - 1 + l)/l, x, y, 2)$ for (b) is
unfolded, \((z, x, y, 2) \rightarrow (z, x, y, 6) \rightarrow (z, x, y, 5) \rightarrow ((z - 1 + l)/l, x, y, 0) \rightarrow ((z - 1 + l)/l, x, y, 1) \rightarrow ((z - 1 + l)/l, x, y, 2)
\). These two paths are edge-disjoint. Therefore Stage 1 is edge-disjoint.

In Stage 2, all nodes do not have identical values of \(x\) and \(y\) at the same unit time on the broadcasting paths except for subpath \(U(0, x, y, 2) \rightarrow (0, x + (m/4), y + (n/4), 2)\) for (c), subpath \(U(0, x, y, 2) \rightarrow (0, x + (m/4), y - (n/4), 2)\) for (d), subpath \(U(0, x, y, 2) \rightarrow (0, x - (m/4), y + (n/4), 2)\) for (e), and subpath \(U(0, x, y, 2) \rightarrow (0, x - (m/4), y - (n/4), 2)\) for (f). Therefore, the paths are edge-disjoint with each other. When subpath \(U(0, x, y, 2) \rightarrow (0, x + (m/4), y + (n/4), 2)\) for (c) is unfolded, \(U(0, x, y, 2) \rightarrow U(0, x + 1, y + 1, 3) \rightarrow \cdots \rightarrow (0, x + (m/4), y + (n/4), 2)\). When subpath \(U(0, x, y, 2) \rightarrow (0, x - (m/4), y - (n/4), 2)\) for (d) is unfolded, \(U(0, x, y, 2) \rightarrow (0, x, y, 1) \rightarrow (0, x, y, 8) \rightarrow (0, x + 1, y + 1, 7) \rightarrow \cdots \rightarrow (0, x + (m/4), y - (n/4), 2)\). When subpath \(U(0, x, y, 2) \rightarrow (0, x - (m/4), y + (n/4), 2)\) for (e) is unfolded, \(U(0, x, y, 2) \rightarrow (0, x, y, 6) \rightarrow (0, x, y, 7) \rightarrow (0, x - 1, y + 1, 8) \rightarrow \cdots \rightarrow (0, x - (m/4), y + (n/4), 2)\). When subpath \(U(0, x, y, 2) \rightarrow (0, x - (m/4), y - (n/4), 2)\) for (f) is unfolded, \(U(0, x, y, 2) \rightarrow (0, x, y, 3) \rightarrow (0, x - 1, y - 1, 2) \rightarrow \cdots \rightarrow (0, x - (m/4), y - (n/4), 2)\). The above four paths are edge-disjoint at the starting node \(U\) under Lemma 2. Therefore, (c), (d), (e), and (f) paths are edge-disjoint under Lemma 1.

\[\square\]

4. Conclusion

Once a new network is designed, various algorithms are proposed for its realization. The algorithms essential for this realization are routing and broadcasting algorithms. Routing algorithms for a 3D PT network have been proposed for design-related studies. In this paper, we proposed an algorithm for one-to-all broadcasting algorithm in an all-port wormhole-routed 3D PT network. This algorithm is based on the divide-and-conquer technique and utilizes the attribute of the 3D PT network that it can be continuously divided into 4 areas. The broadcasting step is \(l/2 + \log_2(n - 1) + 5\) and broadcasting path is disjoint to be effectively used in wormhole-routed 3D PT. This algorithm is used in copying task or data being in a node to another node. Application algorithms such as data migration and task allocation in 3D PT are required to be further studied.

Acknowledgments

This research was supported by Basic Science Research Program through the National Research Foundation (NRF) of Korea funded by the Ministry of Education, Science, and Technology (2012R1A1A4A01014439).

References

[1] C. F. Hsieh, R. C. Chen, and Y. F. Huang, “Applying an ontology to a patrol intrusion detection system for wireless sensor networks,” International Journal of Distributed Sensor Networks. In press.

[2] G. Thatte and U. Mitra, “Sensor selection and power allocation for distributed estimation in sensor networks: beyond the star topology,” IEEE Transactions on Signal Processing, vol. 56, no. 7, pp. 2649–2661, 2008.

[3] J. P. Muñoz-Gea, P. M. Lopez, J. M. Sanahuja, and J. G. Haro, “Design and implementation of a P2P communication infrastructure for WSN-based vehicular traffic control applications,” Journal of Systems Architecture, vol. 59, no. 10, pp. 923–930, 2013.

[4] I. F. Akyildiz, X. Wang, and W. Wang, “Wireless mesh networks: a survey,” Computer Networks, vol. 47, no. 4, pp. 445–487, 2005.

[5] R. P. Liu, G. Rogers, and S. Zhou, “Honeycomb architecture for energy conservation in wireless sensor networks,” in Proceedings of the Global Telecommunications Conference, December 2006.

[6] J.-H. Seo, “Three-dimensional Petersen-torus network: a fixed-degree network for massively parallel computers,” Journal of Supercomputing, vol. 64, no. 3, pp. 987–1007, 2013.

[7] D. Ergu, G. Kou, Y. Peng, Y. Shi, and Y. Shi, “The analytic hierarchy process: task scheduling and resource allocation in cloud computing environment,” Journal of Supercomputing, vol. 64, no. 3, pp. 835–848, 2013.

[8] M. Li and B. Yang A, “Survey on topology issues in wireless sensor network,” in Proceedings of the International Conference on Wireless Networks., pp. 503–509, Las Vegas, Nev, USA, 2006.

[9] Z. Shen, “A generalized broadcasting schema for the mesh structures,” Applied Mathematics and Computation, vol. 186, no. 2, pp. 1293–1310, 2007.

[10] S. M. Hedetniemi, S. T. Hedetniemi, and A. L. Liestman, “A survey of gossiping and broadcasting in communication networks,” Networks, vol. 18, no. 4, pp. 319–349, 1988.

[11] L. M. Ni and P. K. McKinley, “A survey of wormhole routing techniques in direct networks,” Computer, vol. 26, no. 2, pp. 62–76, 1993.

[12] S. Saini, R. Giotti, B. T. N. Gunney et al., “Performance evaluation of supercomputers using HPCC and IMB Benchmarks,” Journal of Computer and System Sciences, vol. 74, no. 6, pp. 965–982, 2008.

[13] P. R. Nuth and W. J. Dally, “The J-Machine network,” in Proceedings of the international conference on computer Design, pp. 420–423, 1992.

[14] S. L. tillevik, “The Touchstone 30 Gigaflop DELTA prototype,” in Proceedings of the 6th Distributed Memory Computing Conference, pp. 671–677, 1991.

[15] Y.-J. Tsai and P. K. McKinley, “A broadcast algorithm for all-port wormhole-routed torus networks,” IEEE Transactions on Parallel and Distributed Systems, vol. 7, no. 8, pp. 876–885, 1996.

[16] S.-K. Lee and J.-Y. Lee, “Optimal broadcast in α-port wormhole-routed mesh networks,” in Proceedings of the International Conference on Parallel and Distributed Systems, pp. 109–114, December 1997.

[17] S.-Y. Wang and Y.-C. Tseng, “Algebraic foundations and broadcasting algorithms for wormhole-routed all-port tori,” IEEE Transactions on Computers, vol. 49, no. 3, pp. 246–258, 2000.

[18] Z. Shen, “An optimal broadcasting schema for multidimensional mesh structures,” in Proceedings of the ACM Symposium on Applied Computing, pp. 1019–1023, March 2003.

[19] S. Latifi, M. H. Lee, and P. K. Srimani, “Wormhole broadcast in hypercubes,” Journal of Supercomputing, vol. 15, no. 2, pp. 183–192, 2000.
[20] J. L. Chen, Y. Xiang, and S. Yao, “Optimum broadcasting algorithm in (n, k)-star graphs using spanning trees,” in *Network and Parallel Computing*, Lecture Notes in Computer Science, pp. 220–230, 2007.

[21] G. Chartrand and R. J. Wilson, “The Petersen graph,” in *Graphs and Applications*, F. Harary and J. S. Maybee, Eds., pp. 69–100.
