Nearly Optimal Algorithms for Broadcast on d-Dimensional All-Port and Wormhole-Routed Torus

Jyh-Jong Tsay and Wen-Tsong Wang
Dept. of Computer Science and Information Engineering
National Chung Cheng University
Chiayi 621, Taiwan, R.O.C.
Email: tsay@cs.ccu.edu.tw

Abstract

In this paper, we present nearly optimal algorithms for broadcast on a d-dimensional $n \times n \times \ldots \times n$ torus that supports all-port communication and wormhole routing. Let $T(n)$ denote the number of communication steps performed to broadcast a message. We present algorithms that achieve the following performance: (i) $T(n) = d\lceil\log_{2d+1} n\rceil + 1$ when n is odd, and (ii) $T(n) = d\lceil\log_{2d+1}(n-1)\rceil + \lceil d/2\rceil + 1$ when n is even. The lower bound is $\lceil d\log_{2d+1} n\rceil$. Our algorithm is the first one that works for arbitrary n, and is optimal up to the constant term. Previous algorithms with comparable performance only work for the case that either d is 2 or 3, or n is a power of $2d+1$. For special cases such as $d = 2$ or 3, we can improve the algorithm so that $T(n) = d\lceil\log_{2d+1} n\rceil$ for any n.

1 Introduction

Broadcast, in which a source processor broadcasts a message to all other processors, is one of the most fundamental collective communications, and is often used for synchronization, initialization, diagnosis, or algorithm execution. In this paper, we study the problem of performing broadcast on a d-dimensional $n \times n \times \ldots \times n$ torus that supports wormhole routing and all-port communication. Notice that torus is one of the most important architectures that have attracted intensive attention and have been used for building parallel computers. A number of recent research have aimed to develop optimal algorithms for collective communications on torus with wormhole and all-port routing [11, 1, 8, 10, 13, 14, 15]. Let $T(n)$ denote the number of communication steps performed by a communication algorithm. The main contribution of this paper is in developing a broadcast algorithm on a d-dimensional $n \times n \times \ldots \times n$ torus, with $T(n) = d\lceil\log_{2d+1} n\rceil + 1$ when n is odd. When n is even, the algorithm can be modified to perform broadcast with $T(n) = d\lceil\log_{2d+1}(n-1)\rceil + \lceil d/2\rceil + 1$. The lower bound is $\lceil d\log_{2d+1} n\rceil$. Notice that our algorithm is optimal upto the constant term. There are several recent research presenting algorithms with comparable performance. In [14], Tseng presents an algorithm that takes $2\lceil\log_5 n\rceil$ on a $n \times n$ torus. He also proposes an approach to generalize his algorithm to run on a $n_1 \times n_2$ torus with time bound $\lceil\log_5 n_1\rceil + \lceil\log_5 \frac{n_1}{2}\rceil + \lceil\log_5 \frac{n_2}{n_1}\rceil + 3$. In [15], Tseng and Wang presents an algorithm that takes $3\lceil\log_7 n\rceil + 2$ on a 3-dimensional $n \times n \times n$ torus. Both the algorithms in [14] and [15] use only dimensional order routing. In [8], Park and Choi present the first algorithm that are efficient for arbitrary dimension d. Although the algorithm presented in [8] achieves the optimal bound, it works only for the case that n is a power of $2d+1$, and requires arbitrary routing capability. Our algorithm is the first one that work for arbitrary d and n, and requires simple routing that is an extension of the dimensional order routing. Moreover, when $d = 2$ or 3, our algorithm can be further improved so that $T(n) = d\lceil\log_{2d+1} n\rceil$ for arbitrary n.

The rest of this paper is organized as follows. Section 2 gives preliminaries. Sections 3, 4 and 5 gives the algorithm for the case that n is a power of $2d+1$. Section 6 sketches modifications of the algorithm for other cases. Section 7 gives further remarks. Due to page limitation, some details are omitted in this paper. All the details can be found in [12].

2 Preliminaries

A d-dimensional $n \times n \times \ldots \times n$ *torus*, denoted as T_{n^d}, is an undirected graph that consists of nodes

$(x_1, \ldots, x_d)$ with $0 \le x_i \le n-1$, $1 \le i \le d$, where x_i is the coordinate of that node in dimension i. Nodes $(x_1, \ldots, x_d)$ and $(y_1, \ldots, y_d)$ are connected by an edge iff there exists a k, $1 \le k \le d$, so that $|x_k - y_k| = 1$ or $n-1$, and $x_i = y_i$, for $\forall i \neq k$ and $1 \le i \le d$.

In this paper, we consider the problem of performing broadcast in torus that supports all-port communication and wormhole routing. In all-port communication, every node can send/receive a message at each of its communication links at every communication step. In wormhole routing, a message is partitioned into flits that are pipelined over the communication path [2, 7]. We also assume the routing always finishes going one dimension before turning to another dimension as in dimension order routing. However, we assume a path can start at any dimension, and then follow the cyclic order of the dimensions until the destination is reached. Namely, assume the coordinates of a source node and a destination node differ at dimensions $i_1, i_2, \ldots, i_k$ with $i_1 < i_2 < \ldots < i_k$. If the routing starts at dimension i_j, $1 \le j \le k$, the subsequent dimensions are $i_{(j+1)\bmod k}$, $\ldots$, $i_{(j+k-1)\bmod k}$. In each dimension, the direction of a path can be either *positive*, i.e. coordinate-increasing, or *negative*, i.e. coordinate-decreasing. We assume a path must follow the same direction in all dimensions. We call this type of routing as *cyclic dimension order*. In cyclic dimension order routing, a path is determined by a starting dimension and a direction.

When there is no edge congestion in the communication paths, the communication latency of wormhole routing is very insensitive to the path length. We thus measure the complexity of a communication algorithm by the number of its communication steps. In each communication step, each processor can compute upto $2d$ destinations, determine a path for each destination, and then send/receive one message at each of its communication links. Such a measure has been used in the analysis of communication algorithms developed on wormhole-routed models [11, 1, 8, 10, 13, 14, 15].

Since at any step, a node can send to at most $2d$ destination nodes, any broadcast algorithm will take at least $\lceil d \log_{2d+1} n \rceil$ communication steps. We thus have the following lower bound.

Lemma 1 *Any algorithm for broadcast on a d-dimensional $n \times n \times \ldots \times n$ all-port and wormhole-routed torus takes at least $\lceil d \log_{2d+1} n \rceil$ communication steps.*

For any two nodes p_1 and p_2 in T_{n^d}, define the *dimension distance*, denoted as $dd(p_1, p_2)$, to be the number of dimensions where their coordinates differ. For example $dd((0, \ldots, 0), (1, \ldots, 1)) = d$, and $dd((1, \ldots, 1, 0), (1, \ldots, 1, 3)) = 1$. In this paper, we will frequently partition T_{n^d} into n^h, $0 \le h \le d$, disjoint $(d-h)$-dimensional subtori by fixing the coordinates in the last h dimensions. We thus, in this paper, define a *$(d-h)$-dimensional subtorus* to be the subgraph in T_{n^d} induced by all the nodes with the same coordinates in the last h dimensions, i.e. by all nodes $(x_1, \ldots, x_{d-h}, c_1, \ldots, c_h)$, where $0 \le x_i \le n-1$, $1 \le i \le d-h$, and $c_1, \ldots, c_h$ are some fixed constants. Given a node $s = (x_1, \ldots, x_{d-h}, c_1, \ldots, c_h)$ in a $(d-h)$-dimensional subtorus, and a vector $v = (v_1, \ldots, v_{d-h}, 0, \ldots, 0)$ with $v_i \in \{1, -1\}$, $1 \le i \le d-h$, the *diagonal* generated by s and v, denoted as $L(s, v)$, consists of the sequence of nodes s, $s+v$, $s+2v$, $\ldots$, $s+(n-1)v$, where $s+mv = (\gamma(x_1+mv_1), \ldots, \gamma(x_{d-h}+mv_{d-h}), x_{d-h+1}, \ldots, x_d)$, $0 \le m \le n-1$, and $\gamma(x) = x \bmod n$ is the *modulo* over n.

Consider a diagonal $L(s, v) = s_0, s_1, \ldots, s_{n-1}$ with $s_i = s + iv$. Let $S = s_{i_1}, s_{i_2}, \ldots, s_{i_k}$ be a subsequence of nodes in $L(s, v)$. Define the *gap* between s_{i_j} and $s_{i_{(j+1)\bmod k}}$ to be $(i_{(j+1)\bmod k} - i_j + 1) \bmod n$ that denotes the number of nodes in $L(s, v)$ that lie between s_{i_j} and $s_{i_{(j+1)\bmod k}}$. The following gives a recursive scheme to grow a subsequence so that the gap between consecutive nodes can differ by at most 1.

Lemma 2 *Let $S_1 = s_{i_1}, \ldots, s_{i_k}$ be a subsequence of nodes in $L(s, v)$ so that the gap between two consecutive nodes in S_1 can differ by at most 1. We can form a larger subsequence S_2 by adding c new nodes between every two consecutive nodes in S_1 so that the gap between any two consecutive nodes in S_2 can differ by at most 1.*

proof: We consider the case when the gap between any two consecutive nodes in S_1 can be either l or $l-1$. Consider the following four integers $l_1 = \lceil \frac{l}{c+1} \rceil$, $l_2 = \lfloor \frac{l}{c+1} \rfloor$, $l_3 = \lceil \frac{l-1}{c+1} \rceil$, and $l_4 = \lfloor \frac{l-1}{c+1} \rfloor$. Let p_1 and p_2 be two consecutive nodes in S_1. WLOG, assume the gap between p_1 and $p-2$ is l. Let $m = l \bmod (c+1)$. We will use l_1 as the gap to take the first m nodes, and l_2 as the gap to take another $c-m$ nodes, starting at the position of p_1. Similarly, when their gap is $l-1$, we use l_3 and l_4 as gaps to take c nodes between them. Since l_1 and l_4 can differ by at most 1, the gap between any two consecutive nodes in S_2 can differ by at most 1. □

In each phase of our algorithms that will be presented later, we will select a set of diagonals as destinations, and run $\lceil \log_{2d+1} n \rceil$ steps to send the message to nodes in the selected diagonal. Above lemma gives us a scheme to take nodes as destinations in every step so that the destinations are distributed evenly in each diagonal.

3 Outline of the Algorithm

To simplify the discussion, we will first present an algorithm for the case that $n = (2d+1)^r$ for some positive integer r. We will explain later in Section 7 how to modify the algorithm for other cases. WLOG, we assume that node $(0, \ldots, 0)$ is the source of the broadcast operation.

The algorithm consists of d phases. After phase h, $1 \le h \le d$, the nodes that already receive the message will be distributed in the n^h $(d-h)$-dimensional subtori so that each subtorus has exactly one such node. We do this by forwarding the message to the nodes in the *main diagonal*, i.e. the diagonal consists of nodes $(i, \ldots, i)$, $0 \le i \le n-1$, in the first phase. In phase h, $2 \le h \le d-1$, we will select a diagonal in each of the n^{h-1} $(d-h+1)$-dimensional subtori as destinations, and then forward the message to all the nodes in the selected diagonals. The message is finally forwarded to all the nodes in the last phase. In any communication step, we say that a node is a *source* node if it already receives the message, and a node is a *destination* node if it is selected to receive the message in that step. The main difficulty in developing the algorithm is how to select the diagonals, and to assign destinations to each source node so that, in every communication step, every source node can forward the message to $2d$ new destinations with congestion-free paths.

Let $D_{h,k}$, $1 \le h \le d$ and $0 \le k \le r$ denote the set of all nodes that have received the message after communication step k of phase h. Notice that $D_{1,0} = \{(0, 0, \ldots, 0)\}$, and $D_{r+1,0}$ will consist of all the nodes in the torus. In the remainder of this paper, we will explain how to perform each of phases 1 to $d-1$ in r steps, and the final phase in $r+1$ steps, when $n = (2d+1)^r$. Therefore, the time bound of above algorithm will be $dr+1$.

4 Broadcast on the Main Diagonal

In this section, we explain how to broadcast the message from source node $(0, \ldots, 0)$ to all the other nodes in the main diagonal, i.e. nodes $(i, \ldots, i)$ for $1 \le i \le n-1$. This will be done in $r(= \lceil \log^n_{2d+1} \rceil$ in general) communication steps as follows. In the first step, we will choose $2d$ destinations for source node $(0, \ldots, 0)$. In each subsequent steps, we will choose $2d$ destinations from nodes between two consecutive source nodes, using $l_k = \frac{n}{(2d+1)^k}$ as the gap. In other words, $D_{1,0}$, $D_{1,1}$, $\ldots$, $D_{1,r}$ are defined as follows. $D_{1,0} = \{(0, \ldots, 0)\}$, and $D_{1,k} = \{(\frac{cn}{(2d+1)^k}, \ldots, \frac{cn}{(2d+1)^k}) | c = 0, \ldots, (2d+1)^k - 1\}$. We will iterate the following steps for $k = 1, \ldots, r$, and after the kth iteration, all the nodes in $D_{1,k}$ will receive the message. Recall that $\gamma(x) = x \bmod n$.

For $k = 1, 2, \ldots, r$, do the following two steps.

1. For each source node $s = (i, i, \ldots, i)$ in $D_{1,k-1}$, select the following $2d$ destination nodes $N_1(s,j) = (\gamma(i + jl_k), \ldots, \gamma(i + jl_k))$, and $N_2(s,j) = (\gamma(i - jl_k), \ldots, \gamma(i - jl_k))$, for $1 \le j \le d$. Note that the union of nodes in $D_{1,k-1}$ and their selected destinations are $D_{1,k}$.

2. For each source node $s = (i, i, \ldots, i)$ in $D_{1,k-1}$, forward the message to $N_1(s,j)$, resp. $N_2(s,j)$, along a path that start at dimension j in positive, resp. negative, direction, for $j = 1, \ldots, d$.

It is clear that every node in $D_{1,k} - D_{1,k-1}$ is selected by exactly one node in $D_{1,k-1}$ as its destination in communication step k. Thus, after above process, all nodes in the main diagonal will receive the message. We show in [12] that there is no edge congestion in every communication step.

Theorem 1 *Phase 1 to broadcast a message from node $(0, 0, \ldots, 0)$ to all the nodes $(i, i, \ldots, i)$, $1 \le i \le n-1$, can be done in r communication steps on a d-dimensional $n \times n \times \ldots \times n$ torus with wormhole and all-port routing, where $n = (2d+1)^r$.*

5 Broadcast from Diagonals to Diagonals

In this section, we explain how to perform phases 2 to $d-1$. Notice that after phase h, $2 \le h \le d-1$, each $(d-h)$-dimensional subtorus will have exactly one node that has received the message. We achieve this in phase h by first choosing n^{h-1} diagonals, one in each of the n^{h-1} $(d-h+1)$-dimensional subtori, and then perform r communication steps to send the message to all the nodes in the selected diagonals. We next explain how to choose diagonals, assign destinations to each source node, and select routing paths, for phase h, $2 \le h \le d-1$.

5.1 Choosing Diagonals

In this section, we explain how to choose diagonals to receive the message in phase h, $2 \le h \le d-1$, and prove some properties of the chosen diagonals that will be used later. Recall that $D_{h,0}$ denotes the set of nodes that are the source nodes in the beginning of phase h. We will choose the diagonals so that the nodes in $D_{h,0}$ are evenly distributed in the torus with one node

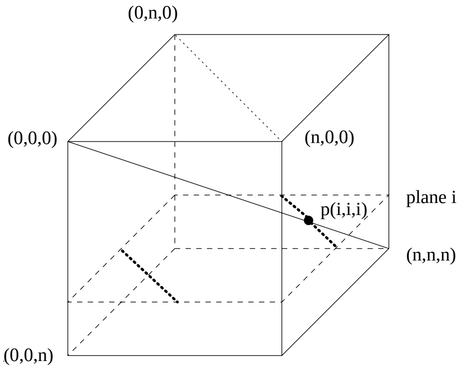

Figure 1. The dotted line is diagonal $L((i,i,i),(1,-1,0))$.

in each $(d-h+1)$-dimensional subtorus. We choose diagonals for phase h as follows. Let $s=(x_1,\ldots,x_d)$ be a node in $D_{h,0}$, and $v=(\underbrace{1,\ldots,1}_{d-h},-1,0,\ldots,0)$ be a vector. Choose the diagonal $L(s,v)$ generated by s and v. Namely, $L(s,v)$ consists of the nodes $(\gamma(x_1+m),\ldots,\gamma(x_{d-h}+m),\gamma(x_{d-h+1}-m),x_{d-h+2},\ldots,x_d)$, $0\le m\le n-1$. (Recall that $\gamma(x)=x \bmod n$.) Figure 1 shows one diagonal chosen in the first step of phase 2 on a 3-dimensional torus. The following lemma can be derived from the way we choose the diagonals.

Lemma 3 *1.* $D_{1,0}=\{(0,0,\ldots,0)\}$

2. $D_{2,0}=\{(k,k,\ldots,k)|0\le k\le n-1\}$

3. $D_{h,0} = \{(\gamma(x_1+m),\ldots,\gamma(x_{d-h+1}+m),\gamma(x_{d-h+2}-m),x_{d-h+3},\ldots,x_d)|(x_1,x_2,\ldots,x_d)\in D_{h-1,0}$, *and* $0\le m\le n-1\}$, *for* $3\le h\le d$.

Proof: immediate from the way we choose the diagonals. □

Notice that the first $d-h+1$ coordinates of any node in $D_{h,0}$ is the same. We thus can denote a node in $D_{h,0}$ as $(z_0,\ldots,z_0,z_1,\ldots,z_{h-1})$. Recall that a $(d-h+1)$-dimensional subtorus consists of the nodes whose coordinates in the last $h-1$ dimensions are fixed. The following lemma is clear from above discussion.

Lemma 4 *Each* $(d-h+1)$*-dimensional subtorus contains exactly one node in* $D_{h,0}$.

Proof: immediate from the definition of $D_{h,0}$. □

Corollary 1 $|D_{h,0}|=n^{h-1}$

Proof: The lemma is clear as there are n^{h-1} $(d-h+1)$-dimensional subtori. □

Let p_1 and p_2 be two nodes in the torus. Recall that the *dimension distance* $dd(p_1,p_2)$ between p_1 and p_2 be the number of dimensions where their coordinates differ.

Lemma 5 *For any two nodes* p_1 *and* p_2 *in the same diagonal chosen in phase* h, $1\le h\le d-1$, $dd(p_1,p_2)\ge 2$ *when* $d\ge 2$.

Proof: immediate from the definition of diagonal. □

Lemma 6 *For any two different nodes* p_1 *and* p_2 *in* $D_{h,0}$, $2\le h\le d$, $dd(p_1,p_2)\ge 2$, *when* n *is odd and* $d\ge 2$.

Proof: We sketch a proof of this lemma. For details, please see [12]. It is clear that above lemma is true for $h=2$ since $D_{2,0}=\{(k,k,\ldots,k)|0\le k\le n-1\}$. Assume above lemma is true for $D_{h,0}$, $h\ge 2$. Consider two nodes $p_1=(z_0,\ldots,z_0,z_1,\ldots,z_h)$ and $p_2=(y_0,\ldots,y_0,y_1,\ldots,y_h)$ in $D_{h+1,0}$. Assume $dd(p_1,p_2)=1$. By Lemma 4, p_1 and p_2 must be in different $(d-h)$-dimensional subtori. Thus, p_1 and p_2 must differ in one of the last h dimension; otherwise, they will be in the same $(d-h)$-dimensional subtorus. Thus, the first $d-h$ coordinates of p_1 and p_2 must be the same. We will derive contradiction for the following cases (i) $z_1\ne y_1$, and (ii) $z_j\ne y_j$ for some j, $2\le j\le h$.

Assume $z_1\ne y_1$. Since $dd(p_1,p_2)=1$, we have $z_j=y_j$, for all $2\le j\le h$. Thus, p_1 and p_2 are in the same diagonal chosen from a $(d-h+1)$-dimensional subtorus in phase h. By Lemma 5, $dd(p_1,p_2)\ge 2$ as they are in the same chosen diagonal. This contradicts our assumption that $dd(p_1,p_2)=1$.

Assume $z_j\ne y_j$, for some j, $2\le j\le h$. By the way we chose the diagonal in phases $j+1$, …, h, there exists m_j, m_{j-1}, …, m_2 so that $0\le m_q\le n-1$, for $j\ge q\ge 2$, and z_0, z_1, …, z_{j-1} can be derived in the following equations.

$$\begin{aligned} z_{j-1} &= (z_j-m_j) \bmod n \\ z_{j-2} &= (z_j+m_j-m_{j-1}) \bmod n \\ &\vdots \\ z_1 &= (z_j+m_j+m_{j-1}+\ldots+m_3-m_2) \bmod n \\ z_0 &= (z_j+m_j+m_{j-1}+\ldots+m_3+m_2) \bmod n \end{aligned}$$

By algegraic calculation, we show in [12] that above equations imply that $dd(p_1,p_2)\ge 2$ which contradicts our assumption. □

In next subsection, we will use Lemma 6 to select edge disjoint paths in every communication step.

Lemma 7 *Given x_0, $\ldots$, x_{j-1}, and x_{j+1}, $\ldots$, x_h, we can determine a unique x_j so that $(x_0, \ldots, x_0, x_1, \ldots, x_h)$ is a node in $D_{h+1,0}$, for any h, $1 \leq h \leq d-1$.*

Proof: Note that there can not be two such nodes; otherwise, their dimension distance will be 1, and Lemma 6 will be violated. We next show that such a node exists.

When $h = 1$, it is clear that $x_1 = x_0$ since $D_{2,0} = \{(x_0, \ldots, x_0) | 0 \leq x_0 \leq n-1\}$. When $h \geq 2$, assume $(x_0, \ldots, x_0, x_1, \ldots, x_h)$ is a node in $D_{h+1,0}$. By the way we choose nodes in phases 1, 2, $\ldots$, h, there exist m_h, m_{h-1}, $\ldots$, m_2 so that $0 \leq m_q \leq n-1$, for $h \geq q \geq 2$, and we have the following equations.

$$\begin{aligned}
x_{h-1} &= (x_h - m_h) \bmod n \\
x_{h-2} &= (x_h + m_h - m_{h-1}) \bmod n \\
&\vdots \\
x_{j+1} &= (x_h + m_h + m_{h-1} + \ldots + m_{j+3} - m_{j+2}) \bmod n \\
x_j &= (x_h + m_h + m_{h-1} + \ldots + m_{j+2} - m_{j+1}) \bmod n \\
x_{j-1} &= (x_h + m_h + \ldots + m_{j+3} + m_{j+1} - m_j) \bmod n \\
&\vdots \\
x_1 &= (x_h + m_h + m_{h-1} + \ldots + m_3 - m_2) \bmod n \\
x_0 &= (x_h + m_h + m_{h-1} + \ldots + m_3 + m_2) \bmod n
\end{aligned}$$

Given x_h, x_{h-1}, $\ldots$, x_{j+1}, we can derive the values of m_h, m_{h-1}, $\ldots$, m_{j+2} from the first $h-j-1$ equations. Given x_0, x_1, $\ldots$, x_{j-1}, we derive the values of m_2, m_3, $\ldots$, m_{j+1} from the last $j+1$ equations. Given m_2, m_3, $\ldots$, m_h, we can compute the valus of x_j so that x_0, $x_1, \ldots, x_h$ satisfies above equations, and hence that $(x_0, \ldots, x_0, x_1, \ldots, x_h)$ is a node in $D_{h+1,0}$. □

Lemma 7 will be used to choose destinations for every source nodes in next subsection.

5.2 Assigning Destinations

In this section, we explain how each node in $D_{h,k-1}$ determines $2d$ destinations from nodes in $D_{h,k} - D_{h,k-1}$, in communication step k of phase h. Recall that $l_k = n/(2d+1)^k$ is the gap used to choose destinations in step k of phase h. $D_{h,k}$ consists of all nodes $(\gamma(x_0 + cl_k), \ldots, \gamma(x_0 + cl_k), \gamma(x_0 - cl_k), x_1, \ldots, x_{h-1})$, $0 \leq c < (2d+1)^k$, for every node $(x_0, \ldots, x_0, x_1, \ldots, x_{h-1})$ in $D_{h,0}$. In other words, we can define $D_{h,k}$ from $D_{h,k-1}$ as follows. $D_{h,k} = \{(\gamma(x_0 + jl_k), \ldots, \gamma(x_0 + jl_k), \gamma(x_1 - jl_k), x_2, x_3, \ldots, x_h) | (x_0, \ldots, x_0, x_1, x_2, x_3, \ldots, x_h) \in D_{h,k-1}, -d \leq j \leq d\}$.

We say that two nodes in $D_{h,k}$ are in the same *column* if they are nodes in the same diagonal chosen in phase h. In each step k of phase h, we assign a rank to nodes in $D_{h,k}$ according to their positions in their diagonals. Namely, for every node p in $D_{h,k}$, the *rank* of node p, denoted as $rank(p)$, is c iff $p = (\gamma(x_0 + cl_k), \ldots, \gamma(x_0 + cl_k), \gamma(x_0 - cl_k), x_1, x_2, \ldots, x_{h-1})$, where $(x_0, \ldots, x_0, x_1, x_2, \ldots, x_{h-1})$ is a node in $D_{h,0}$, and $0 \leq c < (2d+1)^k$.

We say that two nodes in $D_{h,k}$ are in the same *row* iff they have the same rank in $D_{h,k}$. Let s be a node in $D_{h,k-1}$, and the rank of s in $D_{h,k}$ is c. For any node p in $D_{h,k}$, we say that p and s are in the same *row group* iff the rank of p is in the range $[\gamma(c-d), \gamma(c+d)]$. Note that nodes in $D_{h,k}$ are partitioned into $|D_{h,k-1}| = n^{h-1}(2d+1)^{k-1}$ row groups. In step k of phase h, node s will determine $2d$ destinations from nodes in the same row group, $2(d-h+1)$ of them from the column of s, and $2(h-1)$ of them from other columns. We next explain the details.

Let $s = (\gamma(x_0 + cl_k), \ldots, \gamma(x_0 + cl_k), \gamma(x_0 - cl_k), x_1, x_2, \ldots, x_{h-1})$ be a node in $D_{h,k-1}$ with rank c in $D_{h,k}$. In step k of phase h, node s chooses the $2(d-h+1)$ destinations from $D_{h,k}$ that are in the column of s, and have their ranks in the range $[\gamma(c-d+h-1), \gamma(c-1)]$ or $[\gamma(c+1), \gamma(c+d-h+1)]$. Namely, for $1 \leq j \leq d-h+1$, s chooses the following $N_1(s,j)$ and $N_2(s,j)$ from its column as its destinations.

$$\begin{aligned}
N_1(s,j) = (&\gamma(x_0 + (c+j)l_k), \ldots, \gamma(x_0 + (c+j)l_k), \\
&\gamma(x_0 - (c+j)l_k), x_1, \ldots, x_{h-1}) \\
N_2(s,j) = (&\gamma(x_0 + (c-j)l_k), \ldots, \gamma(x_0 + (c-j)l_k), \\
&\gamma(x_0 - (c-j)l_k), x_1, \ldots, x_{h-1})
\end{aligned}$$

Note that the coordinates of $N_1(s,j)$, resp. $N_2(s,j)$, and s differ only in the first $d-h+1$ dimensions. Thus, node s can send the message to nodes in $N_1(s,j)$ and $N_2(s,j)$ with paths totally within the subtorus containing s.

Node s chooses another $2(h-1)$ destinations as follows. For every j, $1 \leq j \leq h-1$, s will choose nodes $N_3(s,j)$ and $N_4(s,j)$ as destinations so that the ranks of $N_3(s,j)$ and $N_4(s,j)$ are $\gamma(c+d-h+1+j)$ and $\gamma(c-d+h-1-j)$, respectively. Furthermore, the dimension distance $dd(s, N_3(s,j)) = 2$ and $dd(s, N_4(s,j)) = 2$. $N_3(s,j)$ and $N_4(s,j)$ are defined as follows.

Let $s_1 = (x_0, \ldots, x_0, x_1, \ldots, x_{h-1})$ be a node in $D_{h,0}$. Let $y_0 = \gamma(x_0 - (d-h+1+j)l_k)$. For $1 \leq j \leq h-1$, we can determine a unique node $s_2 = (y_0, \ldots, y_0, x_1, \ldots, x_{j-1}, y(j), x_{j+1}, \ldots, x_{h-1})$ in $D_{h,0}$ by Lemma 7. Note that $y(j) \neq x_j$; otherwise, we will have two different nodes s_1 and s_2 in the same $(d-h+1)$ dimensional subtorus as $x_0 \neq y_0$, and violate Lemma 4. $N_3(s,j)$ is the node in the column of s_2 with rank $\gamma(c+d-h+1+j)$. Similarly, for $1 \leq j \leq h-1$, we can derive a node $s_3 = (z_0, \ldots, z_0, x_1, \ldots, x_{j-1}, z(j), x_{j+1}, \ldots, x_{h-1})$ in $D_{h,0}$ with $z_0 = \gamma(x_0 + (d-h+1+j)l_k)$ and $z(j) \neq x_j$. $N_4(s,j)$ is the node in the column of s_3 with rank $\gamma(c-d+h-1-j)$. Let $y(j)$ and $z(j)$, $1 \leq j \leq h-1$, be defined as above discussion. We have the following.

$N_3(s,j) = (\gamma(x_0+cl_k), \ldots, \gamma(x_0+cl_k), \gamma(x_0 - cl_k - 2(d-h+1+j)l_k), x_1, \ldots, x_{j-1}, y(j), x_{j+1}, \ldots, x_{h-1})$
$N_4(s,j) = (\gamma(x_0+cl_k), \ldots, \gamma(x_0+cl_k), \gamma(x_0 - cl_k + 2(d-h+1+j)l_k), x_1, \ldots, x_{j-1}, z(j), x_{j+1}, \ldots, x_{h-1})$

Note that the coordinates of s and $N_3(s,j)$, resp. $N_4(s,j)$, differ in dimensions $d-h+1$ and $d-h+1+j$. Node s can send the message to $N_3(s,j)$ and $N_4(s,j)$ with paths consisting of exactly two edges. Furthermore, under above destination assignment, no distinct nodes in $D_{h,k-1}$ choose the same node as their destination. For details of the proof, please see [12].

5.3 Routing Paths

Let s be a source node in $D_{h,k}$. Node s will send the message to its $2d$ destinations as follows.

1. For $1 \leq j \leq d-h+1$, s sends the message to $N_1(s,j)$, resp. $N_2(s,j)$, along a path that starts with dimension j in positive, resp. negative, direction.

2. For $1 \leq j \leq h-1$, s sends the message to $N_3(s,j)$, resp. $N_4(s,j)$, along a path that starts with dimension $d-h+1+j$ in positive, resp. negative, direction.

Lemma 8 *There is no edge congestion among the paths used to send the messages in the same communication step of above algorithm.*

Proof: Consider the paths for two distinct source-destination pairs s_1, d_1 and s_2, d_2 in step k of phase h. Note that $d_1 \neq d_2$. Suppose P_1 is the path from s_1 to s_2, and P_2 is the path from s_2 to d_2. Assume P_1 and P_2 share some edge in the same direction. We say that a *local edge* is an edge connecting two nodes in the same subtorus, and a *global edge* is an edge connecting two nodes in two different subtori. We classify the two paths into the following cases.

Case 1. Both P_1 and P_2 use only local edges. Assume P_1 and P_2 share some edge in the same direction. We have all the edges in P_1 and P_2 are in the same $(d-h+1)$-dimensional subtorus, and s_1, d_1, s_2, and d_2 are in the same $(d-h+1)$-dimensional subtorus. We thus can show that P_1 and P_2 does not share any edge by a proof similar to that in lemma ??.

Case 2. Both paths use one global edge and one local edge. WLOG, assume q_1, resp. q_2, is the intermediate node in P_1, resp P_2. Assume P_1 and P_2 share the same global edge in the same direction. We have $s_1 = s_2$, and $q_1 = q_2$. This is impossible since all the paths originating from s in the positive direction start at different dimensions. Assume P_1 and P_2 share the same local edge. We have $dd(d_1, d_2) \leq 1$. This impossible since $dd(d_1, d_2) \geq 2$ as proved in Lemma 6. Therefore, P_1 and P_2 does not share any edge in the same direction.

Case 3. One path consists of only local edges, and the other consists of a global edge and a local edge. WLOG, assume P_1 consists of only local edges, and P_2 consists of a global edge and a local edge. Let q be the intermediate node in P_1. The common edge of P_1 and P_2 must be the local edge (q, d_1) of P_1, which is an edge along dimension $d-h+1$. Since d_1 and d_2 are in the same subtorus, and hence in the same diagonal, d_1 and d_2 can be denoted as $(x_0, \ldots, x_0, x_1, c_1, \ldots, c_{h-1})$ and $(y_0, \ldots, y_0, y_1, c_1, \ldots, c_{h-1})$ with $x_0 \neq y_0$ and $x_1 \neq y_1$. Since $x_0 \neq y_0$ and $x_1 \neq y_1$, it is impossible for P_2 to pass through d_1. Thus, edge (q, d_1) is not in P_2. Therefore, P_1 and P_2 does not share the same edge in the same direction.

□

Theorem 2 *Phase h, $0 \leq h \leq d-1$, can be done in r communication steps on a d-dimensional $n \times n \times \ldots \times n$ torus with all-port and wormhole routing when $n = (2d+1)^r$.*

6 The Final Phase

Note that after phase $(d-1)$, each 1-dimensional subtorus, referred as a column, will contains exactly one node that has received the message. In the first step of phase d, we will first send the message to nodes in the following set: $S = \{(\gamma(\sum_{i=2}^{d} x_i), x_2, \ldots, x_d) | 0 \leq x_i \leq n-1, 2 \leq i \leq d\}$. It is clear it can be done in one communication step. Nodes in S has the following property.

Lemma 9 *For any j, $2 \leq j \leq d$, the nodes in S that are in the same x_1x_j-plane form a diagonal in that plane.*

Proof: Consider a x_1x_j-plane that consists of all nodes $(x_1, \ldots, x_d)$ with $x_2 = c_2, \ldots, x_{j-1} = c_{j-1}$, $x_{j+1} = c_{j+1}, \ldots, x_d$ where $c_2, \ldots, c_{j-1}, c_j, \ldots, c_d$ are some constants. Let $c = \gamma(c_2 + \cdots + c_{j-1} + c_j + \cdots + c_d)$. Consider a node $(y_1, \ldots, y_d)$ of S in that plane. We have $y_1 = \gamma(\sum_{i=2}^{d} y_i) = \gamma(y_j + c)$. Thus, all the nodes of S in that plane form a diagonal. □

Let $l_k = n/(2d+1)^k$, and $S_0 = S$. Define S_1, S_2, ..., S_r as follows: $S_k = \{(\gamma(x_1 + cl_k), x_2, \ldots, x_d) | (x_1, \ldots, x_d) \in S, 0 \leq c < (2d+1)^k\}$. We say that the *rank* of node $(\gamma(x_1 + cl_k), x_2, \ldots, x_d)$ in S_k is c. By an argument similar to that used in the proof of Lemma 9, we can prove the following lemma.

Lemma 10 *For any j, $2 \leq j \leq d$, the nodes in S that are in the same $x_1 x_j$-plane, and have the same rank form a diagonal in that plane.*

Note that S_{k+1} can be defined from S_k as follows: $S_{k+1} = \{(\gamma(x_1 + jl_k), x_2, \ldots, x_d)| (x_1, x_2, \ldots, x_d) \in S_k$, and $, -d \leq j \leq d\}$. Let $s = (x1, \ldots, x_d)$ be a node in S_k, we use $G^+(s,i)$, resp. $G^-(s,i)$, to denote the node $(\gamma(x_1 + il_k), x_2, \ldots, x_d)$, and resp. $(\gamma(x_1 - il_k), x_2, \ldots, x_d)$, for $1 \leq i \leq d$. We say that node $G^+(s,i)$, resp. $G^-(s,i)$ is the ith destination generated from s in the positive, resp. negative, direction. Note that S_{k+1} consists of the union of S_k and all the destinations generated from nodes in S_k. We do the following operations in the final phase.

1. For $k = 1, 2, \ldots, r$, do the following steps.
   (a) For each node s in S_{k-1}, choose $2d$ destinations from nodes in $S_k - S_{k-1}$ as follows. For each dimension j, along the positive direction of dimension j, choose the jth node, denoted as $N_1(s,j)$, that are in $S_k - S_{k-1}$, and similarly, choose the jth node, denoted as $N_2(s,j)$, along the negative direction. Note that the coordinates of s and $N_1(s,j)$, resp. $N_2(s,j)$, differ only in dimension j.
   (b) For $1 \leq j \leq d$, each node s in S_{k-1} forwards the message to $N_1(s,j)$ along the positive direction, and to $N_2(s,j)$ along the negative direction.

The following lemma gives the correctness of above algorithm.

Lemma 11 *In iteration k of above algorithm, every node in $S_k - S_{k-1}$ will receive the message from exactly one node in S_{k-1}. Furthermore, there is no edge congestion in every step.*

Proof: The proof is based on the following observation. Consider all the nodes in S_k that are in the same $x_1 x_j$-plane. Note that in that $x_1 x_j$-plane, the source nodes will form $(2d+1)^{k-1}$ diagonals, and the destination nodes will form $2d(2d+1)^{k-1}$ diagonals. Furthermore, the destination diagonals are distributed so that there are $2d$ destination diagonals between every two consecutive source diagonals. □

Theorem 3 *Phase d can be done in $r+1$ communication steps on a d-dimensional $n \times n \times \ldots \times n$ torus with all-port and wormhole routing when $= (2d+1)^k$.*

Theorem 4 *In $dr+1$ communication steps, we can broadcast a message from a source node to all other nodes on a d-dimensional torus with wormhole and all-port routing when $n = (2d+1)^r$.*

7 Modifications for Other Cases

In this section, we discuss how to modify our algorithms to handle the case when n is not a power of $2d+1$. We will first discuss the case when n is odd. When n is even, we will first run the algorithm on a $(n-1) \times (n-1) \times \ldots \times (n-1)$ subtorus, and then perform an extra phase to forward the message to nodes not in the subtorus.

When n is odd, it is possible that n can not be divided by $(2d+1)^k$. Thus, in step k of any phase, we can not simply use $\frac{n}{(2d+1)^k}$ as the gap to take $2d$ destinations between two consecutive source nodes. We use the scheme presented in Lemma 2. Since at any step, the gaps between any consecutive source nodes can differ by at most 1, all nodes in the any chosen diagonal will be selected as destinations after $\lceil \log_{2d+1} n \rceil$ steps. Since the destination assignment are based on ranks on each column, we can follow similar approach as presented in Section 5.2 to determine the destinations of every source node. Each source sends the message to its destinations in the same approach as presented in Section 5.3.

Theorem 5 *In $d\lceil \log_{2d+1} n \rceil + 1$ communication steps, we can broadcast a message from a source node to all other nodes on a d-dimensional torus with wormhole and all-port routing, when n is odd.*

When n is even, we will first run the algorithm for odd dimension size on a $(n-1) \times (n-1) \times \ldots \times (n-1)$ subtorus by skipping the nodes with $n-1$ in some of their coordinates. Note that this can be done in $\lceil d \log_{2d+1}(n-1) \rceil + 1$, assuming the wrap-around edge is simulated by a path of length 2 in the original torus. We then perform an extra phase to finish the broadcast, which will be done in $\lceil d/2 \rceil$ steps as follows.

Assume all the nodes $(x_1, \ldots, x_d)$ with $0 \leq x_i \leq n-2$, $1 \leq i \leq d$, already receive the message. We need to forward the message to all the nodes $(x_1, \ldots, x_d)$ with at least one $x_i = n-1$, $1 \leq i \leq d$. We will do this in $\lceil d/2 \rceil$ steps as follows.

Let $p_1 = (x_1, \ldots, x_d)$ be some nodes with exactly one coordinate $x_i = n-1$, and $x_j \neq n-1$ $\forall j \neq i$. Node p will receive the message from the node $s_1 = (x_1, \ldots, x_{i-1}, n-2, x_{i+1}, \ldots, x_d)$ from edge (s_1, p_1) in the positive direction.

Let $p_2 = (x_1, \ldots, x_d)$ be some nodes with exactly two coordinates $x_i = x_j = n-1$, and $x_t \neq n-1$ $\forall t \neq i$ and $t \neq j$. WLOG, assume $i < j$. Node p_2 will receive the message from node $s_2 = (x_1, \ldots, x_{i-1}, 0, x_{i+1}, \ldots, x_{j-1}, 0, x_{j+1}, \ldots, x_d)$ from a path that consists of two edges in the negative direction.

Thus, in one communication step, all the nodes with coordinate $n-1$ in one or two dimensions will receive the message. Assume all the nodes with coordinate $n-1$ in no more than k, $k \geq 2$, dimensions have received the message. Similarly, in one communication step, we can forward the message to all the nodes with coordinate $n-1$ in $k+1$ or $k+2$ dimensions from nodes with coordinate $n-1$ in $k-1$ or k dimensions, considering subproblems on $(d-m)$-dimensional subtori with coordinate $n-1$ in m dimensions. Thus, the extra phase can be done in $\lceil d/2 \rceil$ communication steps.

Theorem 6 *In $d\lceil \log_{2d+1} n \rceil + \lceil d/2 \rceil + 1$ communication steps, we can broadcast a message from a source node to all other nodes on a d-dimensional torus with wormhole and all-port routing, when n is even.*

8 Further Remarks

In this paper, we first present an algorithm for performing broadcast on a d-dimensional $n \times n \times n \ldots \times n$ torus that supports all-port and wormhole routing with cyclic dimension order, when n is a power of $2d+1$. We then explain how to modify the algorithm to handle the case that n is not a power of $2d+1$. Our algorithms are optimal up to the constant term for arbitrary dimension size n. Previous algorithms that achieves comparable performance only work for the case that either $d = 2$ [14] or 3 [15], or n is a power of $2d+1$ [8]. When $d = 2$ or 3, our algorithms can be further improved to achieve the performance that mathes the current best bound achieved in [14] and [15]. Currently, we are trying to improve the performance of our algorithm especially for the case that n is even. When $d = 3$, we are able to eliminate the extra phase for even n. We expect to eliminate the extra phase for arbitrary dimensions in the future.

References

[1] C. Calvin, S. Perennes, and D. Trystram. All-to-all Broadcast in Torus with Wormhole-like Routing. *SPDP*, pp.130-137,1995.

[2] W.J. Dally and C.L. Seitz. The Torus Routing Chip, *JPDC*, 1(3):187-196, 1986.

[3] J.T. Draper and J. Ghosh. Multipath E-Cube Algorithms (MECA) for Adaptive Wormhole Routing and Broadcasting in k-ary n-cubes, *IPPS*, pp. 407-410, 1992.

[4] C.T. Ho and M.T. Raghunath. Efficient Communication Primitives on Hypercubes. *Journal of Concurrency: Practice and Experience*, 4(6):427–457, Sep. 1992.

[5] P.K. Mckinley, H. Xu, A.H. Esfahanian,and L.M. Ni. Unicast-Based Multicast Communication in Wormhole-Routed Direct Networks. *ITPDS*, Vol.5,No.12, pp.1254-1265, Dec. 1994.

[6] P.K. Mckinley, Y.J. Tsai, and D. Robinson. Collective Communication in Wormhole-Routed Massively Parallel Computers. *Computers*,Vol.28,No.12, pp.39-50, Dec. 1995.

[7] L. M. Ni and P. K. McKinley. A survey of wormhole routing techniques in direct networks. *Computers*, Vol:26,No.2,pp.62-76, Feb. 1993.

[8] J.Y.L. Park and H.A. Choi. Circuit-Switched Broadcasting in Torus and Mesh Networks. *ITPDS*, Vol.7,No.2, pp.184-190, Feb. 1996.

[9] J.G. Peters and M. Syska. Circuit-Switched Broadcasting in Torus Networks. *ITPDS*, Vol.7,No.3, pp.246-255, Mar. 1996

[10] D.F. Robinson, P.K. Mckinley, and B.H.C. Cheng. Optimal Multicast Communication in Torus Networks. *ITPDS*, Vol.6,No.10, pp.1029-1042, Oct. 1995.

[11] Y.-J. Tsai and P.K. Mckinley. A Broadcast Algorithm for All-Port Wormhole-Routed Torus Networks. *ITPDS*, Vol.7,No.8, pp.876-885, August. 1996.

[12] J.-J. Tsay and W.-T. Wang. Nearly Optimal Algorithms for Broadcast on d-Dimensional Torus with All-Port and Wormhole Routing. Technical Report, IECS Dept., NCCU, 1997.

[13] Y.-C. Tseng and S.K.S. Gupta. All-to-All Personalized Communication in a Wormhole-Routed Torus. *ITPDS*, Vol.7,No.5, pp.498-505, May. 1996.

[14] Y.-C. Tseng. A Dilated-Diagonal-Based Scheme for Broadcast in a Wormhole-Routed 2D Torus. To appear in *IEEE Trans. on Computers.*

[15] Y.-C. Tseng and S.-Y. Wang. Near-Optimal Broadcast in All-Port Wormhole-Routed 3D Tori with Dimension-Ordered Routing, 1997.