A Lower Bound for Oblivious Dimensional Routing

Andre Osterloh

BTC AG
Escherweg 5
26121 Oldenburg
Germany
osterloh@andre-osterloh.de

Abstract. In this work we consider deterministic oblivious dimensional routing algorithms on d -dimensional meshes. In oblivious dimensional routing algorithms the path of a packet depends only on the source and destination node of the packet. Furthermore packets use a shortest path with a minimal number of bends. We present an $\Omega(kn^{(d+1)/2})$ step lower bound for oblivious dimensional k - k routing algorithms on d -dimensional meshes for odd $d > 1$ and show that the bound is tight.

1 Introduction

Routing is one of the most studied topics in parallel computation. One of the best studied routing problems is the problem where each processor is source and destination of at most k packets, the k - k routing problem. Many different approaches to solve this routing problem have been studied [3]. In [17] the concept of oblivious routing was introduced. In this routing strategy the path of each packet is determined by its source and destination position. This property of oblivious routing algorithms is interesting, since it allows one to design simple and hence practical algorithms. Furthermore, it is of theoretical interest how fast routing problems under such restrictions can be solved. Hence oblivious routing was considered in several publications e.g. [1,2,8,11,14,15]. It was shown that the simplicity of oblivious routing has its costs in the running time. In [6] Kaklamanis, Krizanc, and Tsantilas have shown that every oblivious k - k routing algorithm on a network with N nodes and degree r needs at least $\Omega(k \cdot \sqrt{N}/r)$ steps.

Meshes are very attractive networks due to their simple and regular structure. In a d -dimensional $n \times \dots \times n$ mesh n^d processors are connected by a d -dimensional grid of bidirectional communication links. The degree of a d -dimensional mesh is $2d \in O(1)$ and hence for $d > 1$ an $\Omega(kn^{d/2})$ step lower bound for oblivious k - k routing on these networks exists. For $d = 1$ the bisection bound results in a $\Omega(kn)$ step lower bound. For the case $d = 1$ optimal $O(kn)$ step oblivious k - k routing algorithms that use $O(k)$ buffers are known [3]. For all $d > 1$ asymptotically optimal $O(kn^{d/2})$ step oblivious k - k routing algorithms that use $O(k)$ buffers per node exist [13,10,5].

An interesting question is how simple the paths can be chosen such that the $O(kn^{d/2})$ step bound can be achieved. In [4] *elementary-path* oblivious routing algorithms are introduced and an $\Omega(n^2)$ step lower bound for oblivious elementary-path 1-1 routing algorithms for the case $d = 3$ is shown. In an elementary path routing algorithm every packet uses a shortest path with a minimal number of bends on its way from source to destination. In [10] elementary-path routing algorithms are called dimensional routing algorithms and an asymptotically optimal $O(n)$ step oblivious routing algorithm for the case $k = 1$ and $d = 2$ is presented.

In this work we extend the lower bound of [4] to k - k routing and all odd $d > 1$, i.e., we give a tight lower bound of $\Omega(kn^{(d+1)/2})$ steps for oblivious dimensional k - k routing on d -dimensional $n \times \dots \times n$ meshes. The proof technique used to obtain the lower bound differs from that in [4].

This work is organized as follows. In Section 2, we define d -dimensional meshes, the model and oblivious dimensional routing algorithms. In Section 3, we present the new tight lower bound. Section 4 concludes.

2 Basic Definitions

Throughout this paper we model d -dimensional meshes by directed graphs. The nodes of the graph represent the processors and the edges represent communication links. For a directed graph $G = (V, E)$, and an edge $e = (x, y) \in E$ from x to y we write $src(e) = x$ and $dst(e) = y$. Here *src* stands for source and *dst* for destination. Furthermore, we use the notation src (*dst*) to denote the source (destination) node of packets. For $n \in \mathbf{N}$ we define $[n] = \{0, 1, \dots, n-1\}$ and S_n as the symmetric group¹ of degree n .

2.1 Model of Computation

The processors operate in a synchronous fashion and communicate by sending packets over the unidirectional communication links. We assume that at most one packet can be transmitted by a link from source to destination in one step. In a single step, a processor receives a number of packets that are sent to it via a link in the previous step, performs some amount of internal computation, and sends a number of packets to neighbouring processors. Furthermore processors are able to store an unbounded number of packets in *buffers*.

2.2 Routing Problems

A routing problem on $G = (V, E)$ can be described by a triple $(\mathcal{P}, src, dst)$, where $\mathcal{P}$ is a set of packets, and src, dst are mappings from $\mathcal{P}$ to V . In a routing problem $(\mathcal{P}, src, dst)$ each packet $p \in \mathcal{P}$ is loaded in node $src(p)$ initially and has to be sent to node $dst(p)$.

¹ S_n is the set of all permutations of $[n]$.

A k - k routing problem is a routing problem in which each node is source and destination of at most k packets, i.e., $|src^{-1}(\{v\})| \leq k$ and $|dst^{-1}(\{v\})| \leq k$ for all $v \in V$.

In the literature, a formal definition of oblivious routing algorithms is hard to find. Very often oblivious routing algorithms are described as algorithms where the path of a packet only depends on its source and destination and is independent of other packets, e.g. [7,14,6,10]. We give a formal definition of an oblivious routing algorithm. Before we can do so, we have to define paths.

Definition 1 (path). Let G be a directed graph and $l \in \mathbf{N}$. A path w in G is a (possibly empty) sequence $w = e_0 e_1 \cdots e_{l-1}$ of edges of G such that $dst(e_i) = src(e_{i+1})$ for all $i \in [l-1]$.

If w is a non empty sequence we call w a path from $src(e_0)$ to $dst(e_{l-1})$. If w is an empty sequence we call w a path from x to x for all nodes x of G . $\square$

Definition 2 (oblivious routing algorithm). Let $G = (V, E)$ be a graph and $Path(G)$ be the set of all paths in G . Let $\mathcal{A}$ be a deterministic k - k routing algorithm and let

$$path(\mathcal{A}, (\mathcal{P}, src, dst)) : \mathcal{P} \rightarrow Path(G)$$

be a mapping such that $path(\mathcal{A}, (\mathcal{P}, src, dst))(p)$ is the path on which a packet $p \in \mathcal{P}$ is sent from $src(p)$ to $dst(p)$ by $\mathcal{A}$ when the algorithm solves the k - k routing problem $(\mathcal{P}, src, dst)$.

Algorithm $\mathcal{A}$ is an oblivious k - k routing algorithm on G iff a mapping

$$\pi : V \times V \rightarrow Path(G)$$

exists, such that for all k - k routing problems $(\mathcal{P}, src, dst)$ the following holds:

$$\forall p \in \mathcal{P} : \pi(src(p), dst(p)) = path(\mathcal{A}, (\mathcal{P}, src, dst))(p). \quad \square$$

2.3 Mesh-Connected Networks

One of the most studied networks is the two-dimensional $n \times n$ mesh. In this work we consider its natural generalizations, d dimensional meshes.

Definition 3 (d -dimensional mesh, edge of i -th dimension). The d -dimensional mesh of side length n , denoted by $M_{d,n}$, is a directed graph with node set $[n]^d$ and edge set

$$\{((x_0, \dots, x_{d-1}), (y_0, \dots, y_{d-1})) \mid \forall i \in [d] : x_i, y_i \in [n], \sum_{i=0}^{d-1} |x_i - y_i| = 1\}.$$

An edge $((x_0, \dots, x_{d-1}), (y_0, \dots, y_{d-1}))$ of $M_{d,n}$ where $|x_i - y_i| = 1$ is called an edge of the i -th dimension. $\square$

Since we use directed graphs a bidirectional communication link between two nodes x and y in a mesh is modelled by two directed edges (x, y) and (y, x) .

2.4 Dimensional Routing

In [4] oblivious *elementary-path* routing algorithms are introduced. In an elementary-path routing algorithm the packets use a minimal number of bends on their way from source to destination. Hence in an elementary path algorithm all packets are routed on a shortest path from source to destination. Furthermore, each elementary path in $M_{d,n}$ can be decomposed into at most d subpaths where each subpath consists of edges of (at most) one dimension. The decomposition into paths of different dimension motivates the notation *dimensional-path* algorithms [10] for elementary-path algorithms. In [10] elementary-path routing algorithms are called *weakly-dimensional* routing algorithms.

Before we give a formal definition of oblivious weakly-dimensional and *strongly-dimensional*-path routing algorithms we introduce π -paths for a permutation π . A π -path is a shortest path that can be decomposed into d subpaths p_i such that each p_i only consists of edges of dimension $\pi(i)$.

Definition 4 (π -path in $M_{d,n}$). Let $a, b \in [n]^d$, $\pi \in S_d$ and w be a shortest path from a to b in $M_{d,n}$. Path w is a π -path from a to b iff there exist (possibly empty) paths $p_0, \dots, p_{d-1}$ in $M_{d,n}$ such that

- $w = p_0 p_1 \dots p_{d-1}$,
- $\forall i \in [d]$: All edges of p_i are edges of the $\pi(i)$ -th dimension. □

In Figure 1 we give examples of π -paths. Note that not every shortest path is a π -path for a permutation π . Furthermore, a shortest path could be a π -path for more than one permutation (see Figure 1).

Fig. 1. Some π -paths in $M_{2,6}$. The solid line is a π_1 -path from $(1,1)$ to $(4,5)$ for $\pi_1(0) = 1$ and $\pi_1(1) = 0$. The dashed line is a π_2 -path from $(3,1)$ to $(4,5)$ for $\pi_2 = id$. Finally, the dotted line is a π_1 -path and a π_2 -path from $(5,1)$ to $(5,5)$.

Definition 5 (weakly-dimensional). *An oblivious routing algorithm $\mathcal{A}$ on $M_{d,n}$ is weakly-dimensional iff for all packets p there is a $\pi \in S_d$ such that $\mathcal{A}$ routes p on a π -path from $\text{src}(p)$ to $\text{dst}(p)$.* $\square$

If all packets in a weakly-dimensional routing algorithm use the same permutation π , we get a strongly-dimensional routing algorithm. The well known greedy routing algorithm [9] is an example for an oblivious strongly-dimensional routing algorithm.

Definition 6 (strongly-dimensional). *An oblivious routing algorithm $\mathcal{A}$ on $M_{d,n}$ is strongly-dimensional iff there is a $\pi \in S_d$ such that for all packets p algorithm $\mathcal{A}$ routes p on a π -path from $\text{src}(p)$ to $\text{dst}(p)$.* $\square$

3 Lower Bound

In [4,10,12] oblivious weakly-dimensional 1-1 routing algorithms are considered. In [4] it was shown that any oblivious elementary-path 1-1 routing algorithm on $M_{3,n}$ needs $\Omega(n^2)$ steps. We extend this result to k - k routing on $M_{d,n}$ for all odd d . We use a proof technique different from [4]. Without the restriction to elementary path the lower bound for oblivious routing on $M_{d,n}$ is $\Omega(kn^{d/2})$ [6].

Theorem 1. *For odd d any oblivious and weakly-dimensional k - k routing algorithm on $M_{d,n}$ needs $\Omega(kn^{(d+1)/2})$ steps.*

Proof. For $d = 1$ the result follows by the distance and bisection bound. We assume for the rest of the proof that d is odd and greater than one. Let $\mathcal{A}$ be an oblivious and weakly-dimensional k - k routing algorithm on $M_{d,n}$.

For $t \in [n]^d$, $\pi \in S_d$, we define:

$S(t, \pi) := \{s \in [n]^d \mid \text{Algorithm } \mathcal{A} \text{ sends a packet with source } s \text{ and destination } t \text{ on a } \pi\text{-path from } s \text{ to } t\}.$

For every $t \in [n]^d$ there are n^d possible sources. There are $d!$ permutations $\pi \in S_d$. So

$$\forall t \in [n]^d. \exists \pi \in S_d. |S(t, \pi)| \geq \frac{n^d}{d!}. \quad (1)$$

Since (1) holds for all $t \in [n]^d$ there exists at least one π such that $|S(t, \pi)| \geq n^d/d!$ holds for $n^d/d!$ elements of $[n]^d$. So

$$\exists T \subseteq [n]^d, |T| \geq \frac{n^d}{d!}. \exists \pi \in S_d. \forall t \in T. |S(t, \pi)| \geq \frac{n^d}{d!}. \quad (2)$$

Without loss of generality we can assume in (2) that $\pi = id$. If $\pi \neq id$, we simply permute the dimensions according to π in the rest of the proof. Let $d = 2m + 1$, $m \geq 1$.

For $x \in [n]^m$ we define $M_x := \{(x, y_1, \dots, y_{m+1}) \in [n]^d \mid y_1, \dots, y_{m+1} \in [n]\}$ and $\overline{M}_x := \{(y_1, \dots, y_{m+1}, x) \in [n]^d \mid y_1, \dots, y_{m+1} \in [n]\}$. In $M_{d,n}$ each M_x and $\overline{M}_x$ consists of n^{m+1} nodes. Furthermore there are n^m sets M_x and $\overline{M}_x$, each.

Now we take a T and π that fulfills (2). Since the family $(M_x)_{x \in [n]^m}$ partitions the nodes of $M_{d,n}$ at least one of the n^m sets M_x have a cut of size $(n^d/d!)/n^m = n^{m+1}/d!$ with T . So:

$$\begin{aligned} \exists x \in [n]^m. \exists T \subseteq [n]^d, |T| \geq \frac{n^d}{d!}. \exists \pi \in S_d. \\ (\forall t \in T. |S(t, \pi)| \geq \frac{n^d}{d!}) \wedge |T \cap M_x| \geq \frac{n^{m+1}}{d!}. \end{aligned} \quad (3)$$

We take such x , T and π from (3) and write T_x for $T \cap M_x$. Remember that we assume $\pi = id$. Since the family $(\overline{M_y})_{y \in [n]^m}$ partitions the nodes of $M_{d,n}$ it follows $|S(t, id)| = \sum_{y \in [n]^m} |S(t, id) \cap \overline{M_y}|$ for all $t \in T_x$. From (3) we have $|S(t, id)| \geq n^d/d!$ for all $t \in T_x$. Now assume that a $t' \in T_x$ exists such that there are less than $n^m/(2d!)$ elements $y \in [n]^m$ such that $|S(t', id) \cap \overline{M_y}| \geq n^{m+1}/(2d!)$. For this $t' \in T_x$ it follows $n^d/d! \leq |S(t', id)| = \sum_{y \in [n]^m} |S(t', id) \cap \overline{M_y}| < \frac{n^m}{2d!} n^{m+1} + n^m \frac{n^{m+1}}{2d!} = n^d/d!$. So:

$$\exists x \in [n]^m. \forall t \in T_x. |\{y \in [n]^m : |S(t, id) \cap \overline{M_y}| \geq \frac{n^{m+1}}{2d!}\}| \geq \frac{n^m}{2d!}. \quad (4)$$

Again we take x , T and π from (3) and write T_x for $T \cap M_x$. For $y \in [n]^m$ we define $T_{x,y} := \{t \in T_x : |S(t, id) \cap \overline{M_y}| \geq \frac{n^{m+1}}{2d!}\}$. By (4) for all $t \in T_x$ there are at least $n^m/(2d!)$ elements $y \in [n]^m$ such that $|S(t, id) \cap \overline{M_y}| \geq n^{m+1}/(2d!)$. Furthermore, there are at least $n^{m+1}/d!$ elements in T_x and n^m possibilities for y . Since $\frac{\frac{n^m}{2d!} \cdot \frac{n^{m+1}}{d!}}{n^m} = \frac{n^{m+1}}{2(d!)^2}$ we get:

$$\exists x, y \in [n]^m. |T_{x,y}| \geq \frac{n^{m+1}}{2(d!)^2}. \quad (5)$$

Let $(s_1, \dots, s_d)$ be the source, $(t_1, \dots, t_d)$ be the target, and $(a_1, \dots, a_d)$ be the actual position of a packet. After correcting the first m dimensions of a packet the following holds for the actual position: $a_1 = t_1, \dots, a_m = t_m, a_{m+1} = s_{m+1}, \dots, a_d = s_d$. While correcting dimension $m+1$ only a_{m+1} in the actual position of a packet changes. After correcting $m+1$ dimensions the following holds for the actual position: $a_1 = t_1, \dots, a_{m+1} = t_{m+1}, a_{m+2} = s_{m+2}, \dots, a_d = s_d$.

Now assume that a k - k problem and $x, y \in [n]^m$ exists that fulfills the following property (a):

- (a) After correcting the first m dimensions there are $k \frac{n^{m+1}}{2(d!)^2}$ packets with an actual position $a \in L := \{(x, i, y) \in [n]^d \mid i \in [n]\}$ and while correcting the dimension $m+1$ these packets have to travel a total distance of $\Omega(kn^{m+2})$.

There are at most n edges in L and at most one packet can go over an edge in one time step. So, for correcting the dimension $m+1$ of a k - k routing problem that fulfills (a), algorithm $\mathcal{A}$ needs $\Omega(kn^{m+1})$ steps.

In the rest of the proof we show that a k - k routing problem exists that fulfills (a). We take x, y that fulfill (5). Consider a packet that has destination $t \in T_{x,y}$ and source in $S(t, id) \cap \overline{M_y}$. After the correction of the first m dimensions by algorithm $\mathcal{A}$ the packet is on a node $(x, i, y) \in L$ for an $i \in [n]$ and has to

be transported to a node $(x, j, y) \in L, j \in [n]$. In the case that for a given destination $t = (x, y_1, \dots, y_{m+1}) \in T_{x,y}$, i.e. $j = y_1$, q sources in $\overline{M}_y \cap S(t, id)$ exist there exists at least one source such that a packet has to travel a distance of at least $\lfloor q/(2n^m) \rfloor$ on L , i. e. a source exists such that $|i - j| \geq \lfloor q/(2n^m) \rfloor$. Every $t \in T_{x,y}$ has at least $n^{m+1}/(2d!)$ sources in $S(t, id) \cap \overline{M}_y$ (see (4)). Two elements in $T_{x,y}$ could have the same source, so the number of sources we can choose from is reduced with every choice. So the q in (6) is $n^{m+1}/(2d!) - i$. Since $|T_{x,y}| \geq n^{m+1}/(2(d!))^2$ we have at least $n^{m+1}/(2(d!))^2$ source/destination pairs and the sum in (6) goes from 0 to $n^{m+1}/(2(d!))^2 - 1$. To every found source/destination pair we assign k packets. So we are able to construct a k - k problem such that the total distance the packets have to travel on L is at least

$$k \sum_{i=0}^{\frac{n^{m+1}}{2(d!)^2} - 1} \left\lfloor \frac{\frac{n^{m+1}}{2d!} - i}{2n^m} \right\rfloor \in \Omega(kn^{m+2}). \quad (6)$$

Hence we have proven (a). $\square$

Every oblivious and strongly-dimensional k - k routing algorithm is also an oblivious and weakly-dimensional k - k routing algorithm. Hence

Corollary 1. *For odd d any oblivious and strongly-dimensional k - k routing algorithm on $M_{d,n}$ needs $\Omega(kn^{\frac{d+1}{2}})$ steps.*

The routing algorithm in which all packets use an id -path is a strongly-dimensional oblivious routing algorithm. If we use the furthest destination first priority scheme this algorithm solves a k - k routing problem on $M_{d,n}$ in $O(kn^{\frac{d+1}{2}})$ steps for odd d and in $O(kn^{\frac{d}{2}})$ steps for even d . This can be seen by using the routing lemma of [16] since in our model we allow the processors to store an unlimited amount of packets in buffers. So the lower bound is tight.

4 Conclusion

In this work we presented a tight $\Omega(kn^{(d+1)/2})$ step lower bound for oblivious dimensional k - k routing on d -dimensional meshes for odd $d > 1$. Dimensional $\Omega(kn^{(d+1)/2})$ step oblivious k - k routing algorithms using an unbounded number of buffers exists. For $d > 2$ it is unknown whether the existing lower bounds for dimensional routing can be achieved with algorithms using $O(k)$ buffers.

References

1. Borodin, A., Hopcroft, J.E.: Routing, merging, and sorting on parallel models of computation. In: Proceedings of the 14th Annual ACM Symposium on the Theory of Computing, pp. 338–344 (1982)
2. Borodin, A., Raghavan, P., Schieber, B., Upfal, E.: How much can hardware help routing? In: In: Proceedings of the 25th Annual ACM Symposium on the Theory of Computing, pp. 573–582 (1993)

3. Grammatikakis, M.D., Hsu, D.F., Sibeyn, J.F.: Packet routing in fixed-connection networks: A survey. *Journal of Parallel and Distributed Computing* 54(2), 77–132 (1998)
4. Iwama, K., Miyano, E.: A lower bound for elementary oblivious routing on three-dimensional meshes. *Journal of Algorithms* 39, 145–161 (2001)
5. Iwama, K., Miyano, E.: An $O(\sqrt{N})$ oblivious routing algorithm for 2-D meshes of constant queue-size. *Journal of Algorithms* 41, 262–279 (2001)
6. Kaklamanis, C., Krizanc, D., Tsantilas, T.: Tight bounds for oblivious routing in the hypercube. *Mathematical Systems Theory* 24(4), 223–232 (1991)
7. Krizanc, D.: Oblivious routing with limited buffer capacity. *Journal of Computer and System Sciences* 43(2), 317–327 (1991)
8. Krizanc, D., Peleg, D., Upfal, E.: A time-randomness tradeoff for oblivious routing. In: *Proceedings of the 20th Annual ACM Symposium on the Theory of Computing*, pp. 93–102 (1988)
9. Leighton, F.T.: *Introduction to Parallel Algorithms and Architectures*. Morgan Kaufman, San Mateo (1992)
10. Litman, A., Moran-Schein, S.: Fast, minimal, and oblivious routing algorithms on the mesh with bounded queues. *Journal of Interconnection Networks* 2(4), 445–469 (2001)
11. Meyer auf der Heide, F., Scheideler, C.: Communication in parallel systems. In: Král, J., Bartosek, M., Jeffery, K. (eds.) *SOFSEM 1996*. LNCS, vol. 1175, pp. 16–34. Springer, Heidelberg (1996)
12. Osterloh, A.: Oblivious routing on d -dimensional meshes. In: *Proceedings of the 8th International Colloquium on Structural Information and Communication Complexity*, pp. 305–320. Carleton Scientific (2001)
13. Osterloh, A.: Aspects of k - k routing in Meshes and OTIS Networks. Ph.d. thesis, Technical University of Ilmenau (2002)
14. Parberry, I.: An optimal time bound for oblivious routing. *Algorithmica* 5, 243–250 (1990)
15. Rajasekaran, S., Tsantilas, T.: Optimal routing algorithms for mesh-connected processor arrays. *Algorithmica* 8, 21–38 (1992)
16. Sibeyn, J.F., Kaufmann, M.: Deterministic 1- k routing on meshes. In: Enjalbert, P., Mayr, E.W., Wagner, K.W. (eds.) *STACS 1994*. LNCS, vol. 775, pp. 237–248. Springer, Heidelberg (1994)
17. Valiant, L.G.: A scheme for fast parallel communication. *SIAM Journal on Computing* 11(2), 350–361 (1982)