Randomly Optimized Grid Graph for Low-Latency Interconnection Networks

Koji Nakano, Daisuke Takafuji, Satoshi Fujita
Hiroshima University
1-4-1 Kagamiyama, Higashihiroshima-shi,
Hiroshima, JAPAN 739-8527
{nakano@cs, daisuke@cs, fujita@se}.hiroshima-u.ac.jp

Hiroki Matsutani
Keio University
3-14-1 Hiyoshi, Kohoku-ku,
Yokohama, Kanagawa,
JAPAN 223-8522
matutani@arc.ics.keio.ac.jp

Ikki Fujiwara, Michihiro Koibuchi
National Institute of Informatics
2-1-2 Hitotsubashi, Chiyoda-ku,
Tokyo, JAPAN 101-8430
{koibuchi,ikki}@nii.ac.jp

***Abstract*—In this work we present randomly optimized grid graphs that maximize the performance measure, such as diameter and average shortest path length (ASPL), with subject to limited edge length on a grid surface. We also provide theoretical lower bounds of the diameter and the ASPL, which prove optimality of our randomly optimized grid graphs. We further present a diagonal grid layout that significantly reduces the diameter compared to the conventional one under the edge-length limitation. We finally show their applications to three case studies of off- and on-chip interconnection networks. Our design efficiently improves their performance measures, such as end-to-end communication latency, network power consumption, cost, and execution time of parallel benchmarks.**

I. Introduction

In this work, we tackle a graph problem called the order/degree problem, especially with the subject to edge length in a two-dimensional surface[1]. The order/degree problem with parameters n and K is to find a graph with minimum diameter over all graphs with the number of vertices $= n$ and degree $\leq K$. If two or more graphs take the minimum diameter, a graph with minimum average shortest path length (ASPL) over all graphs with minimum diameter must be found.

Graph structure has been studied in terms of ASPL and diameter. The Moore bound illustrates ideal graphs in terms of path hops, though they may not exist[2]. Given a graph with degree K and diameter i, the number of vertices in the graph is at most $1+K\sum_{j=0}^{i-1}(K-1)^j$. The most famous graph problem that relates to the Moore bound is called degree/diameter problem (DDP). The DDP has been studied for decades and consists in generating the largest possible graph given degree and diameter constraints, striving to approach theoretical upper bounds.

From the engineering point of view, the order/degree problem is radically different from DDP. Although the DDP is theoretically interesting, its application field to computer networks is limited. DDP solutions may not be directly usable for building network topologies in supercomputers, high-end datacenter systems and on-chip multiprocessors, because they are for particular number of compute nodes, whereas the number of nodes in a real system is determined based on practical considerations, e.g., budget. We thus consider that the order/degree problem will be important in the field of interconnection networks.

The tight design constraints of recent on-chip and off-chip interconnection networks make the order/degree problem and its model complex. A short link design is preferred in various interconnection networks, such as network-on-chips (NoCs) and also in supercomputers for better wiring and floorplan[3], [4]. In this context we present (1) the randomly optimized grid graphs that maximize the performance measure, such as diameter and average shortest path length (ASPL) with subject to limited edge length on a grid surface, (2) their theoretical lower bounds of the diameter and ASPL, which prove optimality of our randomly optimized grid graphs, and (3) a diagonal grid layout that significantly reduces the diameter compared with the conventional one under the edge length limitation.

Our main contributions in this work are as follows.
Section III: We propose a heuristic method to build a randomly optimized grid graph that corresponds to the network topologies under tight constraints of the link length on a floorplan.
Sections IV, V and VII: We theoretically illustrate the lower bound of both diameter and ASPL of the grid graph. The diameter and the ASPL of the proposed grid graph is almost optimal in the sense that they are very close to the theoretical lower bounds. We illustrate well-balanced values of degree and maximum cable length.
Section VI: We devise an efficient geometric structure called diagrid to reduce the diameter of the grid graph without increasing degree and maximum edge length. It gives a short Manhattan distance between the farthest nodes.
Section VIII: We provide quantitative comparisons of network topologies through practical case studies. We find that the resulting optimized topologies outperform by 55% on average the counterpart k-ary n-cubes in terms of parallel benchmark performance. Our network design also works well for improving off-chip network power consumption and on-chip parallel application performance.

Background information and related work are discussed in Section II. Section IX concludes with a summary of our findings.

II. Background and Related Work

A. Graphs with Low Diameters and Low ASPLs

The diameter/degree problem (DDP) has been considered for various types of graphs, such as Caylay and circulant versions [2]. Several graphs with tractable and hierarchical structure and good diameter properties have been reported, including the well-known De Bruijn graphs, Kautz graphs and (n,k)-star graphs. The distance to the Moore bound makes the DDP interesting. The DDP best known solutions typically have relatively small numbers of vertices, and are thus far from the Moore bound [2]. For example, most best known solutions for $d > 7$ and $K > 7$ general graphs achieve less than 10%.

Our interests are the order/degree problem that might relate to the DDP: find a graph that has smallest diameter and ASPL given an order and a degree. Its lower bound on the ASPL has been computed from the Moore bound[5]. However, when a grid graph is considered, the problem becomes complex. If the target is a grid graph where each vertex is located in a two-dimensional surface and an upper bound of the edge length is given, its theoretical bound of diameter/ASPL and best known graphs are not resolved and not listed in a catalogue. We provide the solutions of the grid graph in this study.

B. Order/Degree Graph Application to Interconnection Networks

High-performance computing systems with possibly millions of cores is required to achieve low network latency, e.g., 1μs across the system, as well as high bisection bandwidth [6]. To achieve low latency, a topology of switches should thus have low diameter and low average shortest path length (ASPL), both measured in numbers of switch hops for given degree and network size [7]. This fact motivates the application of the order/degree solution graph for the network topology of interconnection networks.

1) Off-Chip Network Topology: A few topologies (e.g., k-ary n-cubes, fat trees, and Dragonfly) have been used to connect compute nodes in most HPC systems. Historically, in a conservative engineering approach, the main use of short cables is preferred for good floor layout in a machine room. The k-ary n-cubes only have short cables (e.g., in K computer[4]). Fat trees and Dragonfly use some long optical inter-cabinet links to have low hop counts exploiting high-radix switches. More aggressively, a random topology, that may have a large number of long links that makes cabling complex, achieves low diameter and low average shortest path length (ASPL)[7].

Note that InfiniBand and Ethernet used in recent commodity interconnection networks have routing tables and support arbitrary topologies. To support arbitrary topologies and their routing, the path calculation costs for deadlock-free topology-agnostic routing[8] and the scalability issues are no longer severe problems[9]. In this context, we can apply our randomly optimized grid graph for those network topologies.

2) On-Chip Network Topology: The k-ary 2-meshes and folded k-ary 2-tori have good layouts that make each link length uniform and short. In the flattened butterfly[10], routers in each row of a conventional butterfly are combined into a single router. It has a large diversity of router degrees for various network sizes and its low-degree case is equivalent to hypercube. When a design allows each router to have multiple cores and slightly longer links, a cost-effective design, such as concentrated mesh, can be used[11]. Random and small-world network topologies are also well considered in on-chip networks[12], because random long links strongly reduce packet path hops. There are some techniques concerning signal propagation delay. Express virtual channels and SMART can bypass router pipelines for certain access patterns so that a flit could reach the destination almost by its signal propagation delay[13]. Elastic buffers and on-chip decentralized routers can optimize energy on wires[14] by distributing the router functions over links.

In this work we compare our randomly optimized grid graph (topology) to those same-degree network topologies in Section VIII.

III. Randomly Optimized K-regular L-restricted Grid Graphs

A grid graph is a graph $G = (V,E)$ such that $V = \{(x,y) \mid 0 \le x,y \le \sqrt{N}-1\}$ is a set of N nodes and E is a set of edges connecting a pair of two distinct nodes in V. We can think that nodes in V are arranged in a 2-dimensional space so that each node (x,y) is located at position (x,y). Let $l(u,v)$ denote the Manhattan distance of two nodes u and v in V, that is, $l(u,v) = |u_x - v_x| + |u_y - v_y|$, where $u = (u_x,u_y)$ and $v = (v_x,v_y)$. In a network with topology represented by a grid graph, the two nodes (u,v) are connected by a communication link of length $l(u,v)$ wired along the grid.

A grid graph $G = (V,E)$ is *L-restricted* if $l(u,v) \le L$ for all edges $(u,v) \in E$. Clearly, in a network with topology represented by an L-restricted grid graph, the length of every communication link is restricted to no more than L. A graph is *K-regular* if every node is connected with K edges. The value of K corresponds to the number of ports equipped with a computer or a network switch.

Let $D(G)$ and $A(G)$ be *the diameter* and *the average shortest path length (ASPL)* of a graph G, respectively. More formally, $D(G)$ and $A(G)$ can be defined using $h_G(u,v)$, the number of edges in the shortest path between nodes u and v in G as follows:

$$\begin{aligned} D(G) &= \max\{h_G(u,v) \mid u,v \in V\} \\ A(G) &= \sum_{u \neq v} h_G(u,v)/(N(N-1)) \end{aligned}$$

Our goal is to find a K-regular L-restricted grid graph G with the minimum diameter $D(G)$. Further, we want to select a graph G with the minimum ASPL $A(G)$ over all graphs with the minimum diameter. We say that *G is better than G'* if $D(G) < D(G')$, or both $D(G) = D(G')$ and $A(G) < A(G')$ hold. In other words, our goal is to find the best or an almost best graph over all K-regular L-restricted grid graphs. Our algorithm for this goal may need to handle an unconnected grid graph, in which the diameter and the ASPL cannot be computed, as an intermediate one. Hence, we extend this definition as follows. When at least one of G and G' are unconnected, we say that *G is better than G'* if $C(G) < C(G')$, where $C(G)$ is the number of connected components of G.

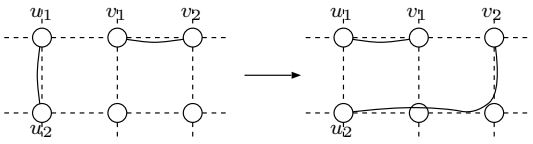


Fig. 2. 2-toggle operation

We will show a randomized algorithm for generating an almost optimal K-regular L-restricted grid graph with small diameter and small ASPL. Our algorithm has three steps as follows:

Step 1: Generate an initial K-regular L-restricted grid graph G.

Step 2: Repeatedly perform *random 2-toggle operation* to generate a random K-regular L-restricted grid graph.

Step 3: Repeatedly perform *random 2-opt operation* to find a K-regular L-restricted grid graph as best as possible.

In Step 1, an initial graph can be generated by connecting two adjacent nodes appropriately. Note that the topology of an initial graph generated in Step 1 is not a big issue. Since the following steps generate a random graph by scrambling the edges, an initial grid graph can be any K-regular L-restricted grid graph.

Step 2 repeats the random 2-toggle operation illustrated in Figure 2. In the random 2-toggle operation, we randomly pick two disjoint edges. In the figure, (u_1,u_2) and (v_1,v_2) are selected. These edges are replaced with two edges (u_1,v_1) and (u_2,v_2). Clearly, this replacement does not change the degree of every node. However, it is possible that the graph thus obtained is not L-restricted, that is, $l(u_1,v_1)>L$ or $l(u_2,v_2)>L$ hold. If this is the case, we undo the replacement. The random 2-toggle operation is repeated for all edges in G. Figure 1 (2) shows a random grid graph obtained when Step 2 terminates. The shortest paths from the node at the top-left corner to the other corners are colored.

In Step 3, the random 2-opt operation is repeated many times. The random 2-opt operation is almost the same as the 2-toggle operation. The difference is cancellation of the replacement. We also undo the random 2-opt operation if the resulting graph is not better than the graph before the operation. Figure 1 (3) shows a graph obtained after Step 3. We can see that the iterative random 2-opt operations can generate a graph with smaller diameter and ASPL. Since the random 2-opt operation is a simple local search for finding a local minimum, we have applied simulated annealing technique, a generic metaheuristic for the optimization, which can be more efficient than a straightforward local search. More specifically, even if the resulting graph obtained by the 2-toggle operations is not better, we do not cancel the replacement with some small probability. We call a graph obtained by our randomized algorithm *a randomly optimized grid graph*.

We should mention that Step 2 can be omitted. An optimized random grid graph can be obtained just by executing Step 1 and Step 3. We execute Step 2 to accelerate the computation. The computation cost of the random 2-opt operation is much larger than that of the random 2-toggle operation, because it must compute both the diameter and the ASPL, which takes $O(N^2K)$ time by breadth first search starting from every node. On the other hand, the random 2-toggle operation can be done in $O(1)$ time and thus the running time of Step 2 is very small. For example, Step 2 runs in less than 0.1 seconds for parameters $K=6$, $L=6$ and $N=30\times 30$, and generates a grid graph with diameter 12 and ASPL 5.7933 using Intel Core i7-4650 CPU. If we omit Step 2, then more than 1,800 iterations of the random 2-opt operations, which takes more than 70 seconds, are performed to obtain a grid graph with the same diameter and ASPL. Hence Step 2 is very helpful to get a good intermediate solution of a grid graph at a small computing cost.

IV. Lower Bounds of the Diameter and ASPL of Grid Graphs

This section shows tight lower bounds of the diameter and the ASPL of grid graphs. Throughout this section, we focus on K-regular L-restricted grid graphs of size $\sqrt{N}\times\sqrt{N}$. We first show the lower bound of the ASPLs for K-regular graphs and for L-restricted grid graphs, separately. After that, we combine them to obtain a tight lower bound for K-regular L-restricted grid graphs.

First, we show the lower bound of the ASPL of a K-regular graph $G=(V,E)$. For any fixed node u, we can partition all nodes in V into groups $V_0,V_1,\ldots$ such that $V_i=\{v\in V \mid$ the shortest path between u and v has i edges$\}$. Clearly, $V_0=\{u\}$ holds. Since u is connected with K edges, V_1 has K nodes. Further, each node in V_1 is connected with K nodes in $V_0\cup V_1\cup V_2$, and one of them is u ($\in V_0$). Hence, it is connected with at most $K-1$ nodes in V_2 and so V_2 has at most $K(K-1)$ nodes. Similarly, each node in V_2 is connected with at most $K-1$ nodes in V_3, and thus V_3 has $K(K-1)^2$ nodes. In general, V_i ($i\geq 1$) has no more than $K(K-1)^{i-1}$ nodes. Let $m(i)$ be the Moore function such that $m(1)=1$ and

$$m(i) \quad = \quad \max(1+\sum_{j=1}^{i} K(K-1)^{j-1}, N) \qquad (1)$$

for all $i>1$. It should be clear that the number of nodes reachable in i hops from u does not exceed $m(i)$. Thus, we have the lower bound of the ASPL A_m^- of a K-regular graph as follows:

$$A_m^- \quad = \quad \sum_{i\geq 1}((m(i)-m(i-1))\cdot i)/(N-1). \qquad (2)$$

Next, we show the lower bound of the ASPL of an L-restricted grid graph. To show the lower bound, we evaluate the ASPL of a grid graph in which all pair of two nodes within distance L are connected. The number of nodes that can be reachable in i hops from a node (x,y) is

$$d_{x,y}(i)=|\{(x',y')\in V \mid d((x,y),(x',y'))\leq i\cdot L\}|. \qquad (3)$$

Figure 3 illustrates the values of $d_{0,0}(i)$ for a 3-restricted grid graph of size 10×10. The lower bound A_d^- of the ASPL of an L-restricted grid graph can be computed by the following formula:

$$A_d^- \quad = \quad \sum_{(x,y)\in V}\sum_{i\geq 1}((d_{x,y}(i)-d_{x,y}(i-1))\cdot i)/N(N-1). \qquad (4)$$

It should be clear that $\max(A_m^-,A_d^-)$ is the lower bound of the ASPL of a K-regular L-restricted grid graph, because it is

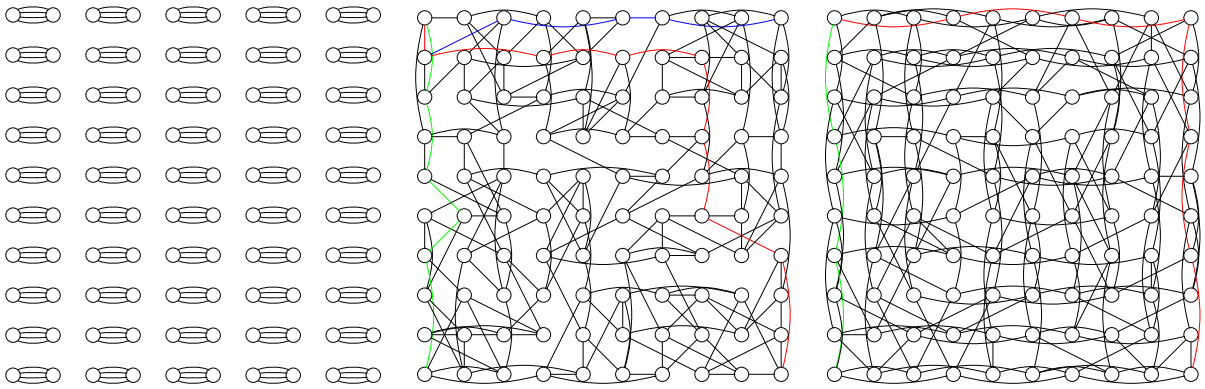


Fig. 1. The topologies of 4-regular 3-restricted grid graphs created at the end of each step. Edges are drawn straight for visibility, although they should be wired along the grid.

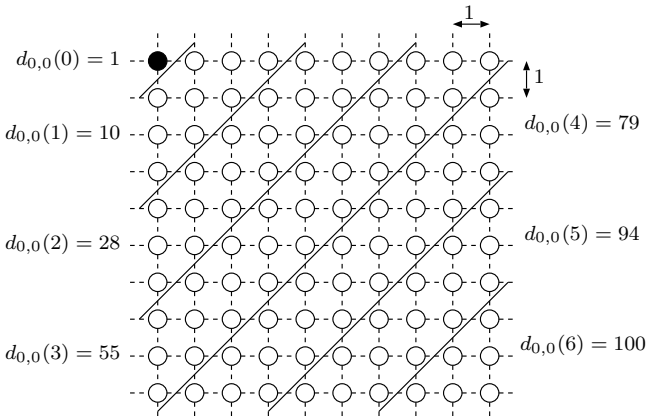


Fig. 3. The values of $d_{0,0}(i)$ for a 3-restricted grid graph of size 10×10.

a K-regular graph as well as an L-restricted grid graph. We can obtain a better, that is, larger lower bound by combining $m(i)$ and $d_{x,y}(i)$. Let $md_{x,y}(i) = \min(m(i), d_{x,y}(i))$. Clearly, the number of nodes reachable in i hops from node (x,y) in a K-regular L-restricted grid graph does not exceed $md_{x,y}(i)$. Thus, we have the lower bound A^- by the following formula:

$$A^- = \sum_{(x,y)\in V} \sum_{i\geq 1} ((md_{x,y}(i) - md_{x,y}(i-1)) \cdot i)/N(N-1).$$

We can also have the lower bound of the diameter using md. Let D^- be the value of i such that $md_{0,0}(i-1) < N$ and $md_{0,0}(i-1) = N$. Since there exists a node v such that the shortest path from $(0,0)$ to v has D^- edges, D^- is the lower bound of the diameter.

Table I shows the values of m, d, and md for a 4-regular 3-restricted grid graph of size 10×10. We can see that $m(i) < d_{0,0}(i)$ for small i and $m(i)$ increases more rapidly than $d_{0,0}(i)$. From the table, we have the diameter lower bound $D^- = 6$. Also, we can obtain the lower bound $A^- = 3.330$ of the ASPL, which is larger than $A^-_m = 3.273$ and $A^-_d = 2.560$. Thus, the diameter of a 4-regular 3-restricted grid graph shown in Figure 1 (3) is optimal. Also, the ASPL is almost optimal; the gap is only $\frac{3.443-3.330}{3.330} \approx 3.4\%$.

TABLE I. THE VALUES OF m, $d_{0,0}$, AND $md_{0,0}$ FOR A 4-REGULAR 3-RESTRICTED GRID GRAPH OF SIZE 10×10

i	0	1	2	3	4	5	6
$m(i)$	1	5	17	53	100	100	100
$d_{0,0}(i)$	1	10	28	55	79	94	100
$md_{0,0}(i)$	1	5	17	53	79	94	100

For later reference, we write $D^-(N,K,L)$, $A^-(N,K,L)$, $A^-_m(N,K)$, and $A^-_d(N,L)$ when we should clarify that the values of D^-, A^-, A^-_m, and A^-_d are for a K-regular L-restricted grid graph of size $\sqrt{N} \times \sqrt{N}$. Also, let $D^+(N,K,L)$ and $A^+(N,K,L)$ be the values of the diameter and the ASPL of optimized grid graph generated by our randomized algorithm. We may also omit N if it is clear from the context.

V. OPTIMALITY OF RANDOMLY OPTIMIZED GRID GRAPHS

In this section, we discuss optimality of the optimized grid graphs that our randomized algorithm generates by comparing the diameter and the ASPL with the lower bounds and show that they are almost optimal. We focus on a K-regular L-restricted grid graph of size 30×30.

We will show that the diameter $D^+(K,L)$ of the optimized grid graph is equal to the theoretical lower bound $D^-(K,L)$ for most of K and L. Table II shows the values of $D^+(K,L)$ and $D^-(K,L)$ for all K ($3 \leq K \leq 12$) and L ($2 \leq L \leq 12$). Since $D^+(7,L), D^+(8,L)$, and $D^+(9,L)$ are the same for every L, these values are written in row $D^+(7\text{–}9,L)$. Similarly, rows $D^+(10\text{–}16,L)$ and $D^-(6\text{–}16,L)$ corresponds to multiple K's that take the same diameters for every L. From the table, we can see that the values of $D^+_{K,L}$ and $D^-_{K,L}$ are almost the same. In particular for large K or for small L, they are equal and randomly optimized grid graphs are diameter-optimal. On the other hand, for small K and large L, $D^+(K,L)$ and $D^-(K,L)$ are not equal. If L is large, we have a lot of choices of edges connecting nodes, and thus, it is very hard to find the best one. Also, if K is small, each edge is used for the shortest paths of many pairs of nodes. Hence, generation of best configuration of edges for small K and large L is difficult.

However, the non-optimal cases are not big issues, because we can select the best pair of K and L to satisfy a given

TABLE II. DIAMETER UPPER BOUND $D^+(K,L)$ AND LOWER BOUND $D^-(K,L)$ OF A K-REGULAR L-RESTRICTED GRID GRAPH OF SIZE 30×30

L	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
$D^+_{3,L}$	29	20	15	12	12	11	11	11	11	11	11	11	11	11	11
$D^-_{3,L}$	29	20	15	12	10	9	9	9	9	9	9	9	9	9	9
$D^+_{4,L}$	29	20	15	12	10	9	8	8	8	8	8	8	8	8	8
$D^-_{4,L}$	29	20	15	12	10	9	8	7	6	6	6	6	6	6	6
$D^+_{5,L}$	29	20	15	12	10	9	8	7	7	6	6	6	6	6	6
$D^-_{5,L}$	29	20	15	12	10	9	8	7	6	6	5	5	5	5	5
$D^+_{6,L}$	29	20	15	12	10	9	8	7	6	6	6	6	6	6	6
$D^+_{7-9,L}$	29	20	15	12	10	9	8	7	6	6	5	5	5	5	5
$D^+_{10-16,L}$	29	20	15	12	10	9	8	7	6	6	5	5	5	4	4
$D^-_{6-16,L}$	29	20	15	12	10	9	8	7	6	6	5	5	5	4	4

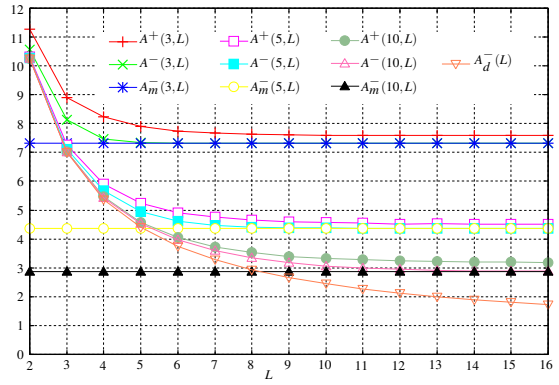


Fig. 4. **ASPL** of grid graphs of size 30×30 for $K=3$, 5, and 10

diameter condition. For example, if diameter 8 is required, we can find from Table II that $K=4$ and $L=8$ must be selected. Also, we can see that $D^-(K,L)\geq 9$ when $K\leq 3$ or $L\leq 7$. Hence, the degree $K=4$ and the maximum edge length $L=8$ are must to attain diameter 8. In other words, our randomly optimized 4-regular 8-restricted grid graph with diameter 8 is optimal in the sense that it is proved that any grid graph with smaller parameters always has diameter larger than 8. If we need a grid graph with diameter 7, then we can select parameters $K=5$ and $L=9$. Unfortunately, the graph obtained by our algorithm with these parameters is not proved optimal. From $D^-(4,9)=8$, there may exist a 4-regular 9-restricted grid graph with diameter 8. Of course, there may not exist such a graph, because the lower bound is not tight. However, we can say that it is almost optimal because plus one degree achieves the optimal diameter graph.

Next, we will discuss the ASPL of grid graphs of size 30×30. Figure 4 shows the upper bounds and the lower bounds for $K=3$, 5, and 10 from $L=2$ to 16. Combining each $A^-_m(3)=7.325$, $A^-_m(5)=4.377$, and $A^-_m(10)=2.878$ with $A^-_d(L)$, we have the lower bounds $A^-(3,L)$, $A^-(5,L)$, and $A^-(10,L)$. We can see that the upper bounds $A^+(3,L)$, $A^+(5,L)$, and $A^+(10,L)$ obtained by our randomly optimized grid graph are very close to these lower bounds. From the figure, we can see that the improvement of the ASPL is saturated for large L. Hence, it makes no sense to select too large L. For example, from the values of $A^+(5,L)$ in Figure 4, it makes no sense to choose $L\geq 10$ when $K=5$, because the ASPL $A^+(5,L)$ for $L\geq 10$ is almost the same as that for $L=9$.

Figure 5 shows the upper bounds and the lower bounds for $L=3$, 5, and 10 from $K=3$ to 16. We can see that it is very similar to Figure 4. Combining the lower bounds $A^-_d(3)=$

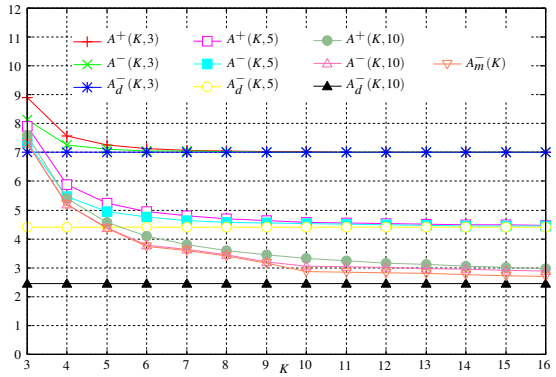


Fig. 5. **ASPL** of grid graphs of size 30×30 for $L=3$, 5, and 10

7.000, $A^-_d(5)=4.401$, and $A^-_d(10)=2.452$ with $A^-_m(K)$, we have the lower bounds $A^-(K,3)$, $A^-(K,5)$, and $A^-(K,10)$. We can see that the upper bounds $A^+(3,L)$, $A^+(5,L)$, and $A^+(10,L)$ are very close to the lower bounds. Also, the improvement is saturated for large L, and thus, we can find appropriate values of L for each K in the same way.

VI. RANDOMLY OPTIMIZED K-REGULAR L-RESTRICTED DIAGRID GRAPHS

This section presents an idea to reduce the diameter of a grid graph without increasing degree K and maximum edge length L. The idea is to use a diagonal grid (i.e., a diagrid) arrangement of nodes and edges.

Suppose that $\sqrt{\frac{n}{2}}\times\sqrt{2n}$ nodes are arranged in $\sqrt{\frac{n}{2}}$ rows such that each row has $\sqrt{\frac{n}{2}}$ nodes and the Euclidean distance of adjacent nodes is $\sqrt{2}$. Each odd rows are slided by $\sqrt{2}/2$. We call a graph arranged in this way *a diagrid graph*. We assume that the distance of two nodes is defined by the Manhattan distance along diagonal directions. For example, the distance of two nodes adjacent in a diagonal direction is 1. The distance of two nodes adjacent in a horizontal direction is 2, although the Euclidean distance of them is $\sqrt{2}/2$. Figure 6 illustrates the node arrangement of a diagrid graph of size 7×14. As the Manhattan distance is defined along slash lines, communication links of a network represented by a diagrid graph are wired along them.

We can generate *a randomly optimized diagrid graph* in three steps almost in the same way as a grid graph. More specifically, an initial diagrid graph is generated (Figure 7 (1)) in Step 1. In Step 2, the random 2-toggle operation is repeated for generating a random diagrid graph (Figure 7 (2)). Finally, Step 3 repeats the random 2-opt operation to obtain a randomly optimized diagrid graph (Figure 7 (3)). We can see that the iterative random 2-opt operation can decrease the diameter and the ASPL significantly.

Let us compare the randomly optimized diagrid graph (Figure 7 (3)) to the randomly optimized grid graph (Figure 1 (3)). Note that the numbers of nodes are almost the same: the diagrid graph has 98 nodes while the grid graph has 100 nodes. The diameter of the diagrid graph is smaller than that of the grid graph, because the maximum distance of two nodes is smaller. That of the diagrid graph of size 7×14 is only 13, while that of the grid graph of size 10×10 is 18. In general, that of the

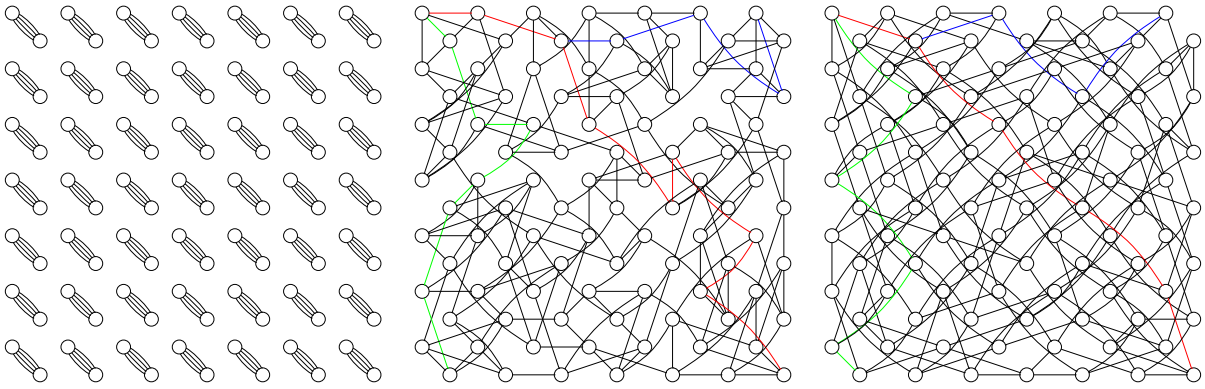


Fig. 7. Illustrating 4-regular 3-restricted diagrid graphs of size 7×14 generated by our algorithm. **Edges are drawn along Euclid for the sake of simplicity, although they are along Manhattan distance.**

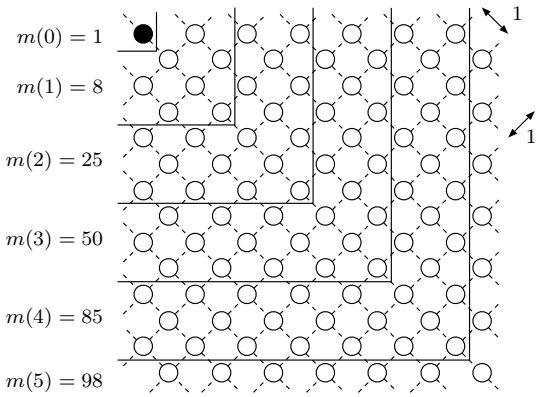


Fig. 6. A diagrid graph of 7×14 and the values of $d_{0,0}(i)$ for 3-restricted

the diagrid graph of size $\sqrt{\frac{n}{2}} \times \sqrt{2n}$ is $\sqrt{2n}-1$, while that of the grid graph of size $\sqrt{n} \times \sqrt{n}$ is $2\sqrt{n}-2$. Thus, the diameter of the diagrid graph may be decreased to

$$\frac{\sqrt{2n}-1}{2\sqrt{n}-2} \approx \frac{\sqrt{2}}{2} \approx 70.7\%$$

for enough large n. On the other hand, the ASPL of two graphs are almost the same, because the average distances of them are almost equal. Actually, the average distance of nodes of a 10×10 grid graph is 6.667, while that of a 7×14 diagrid graph is 6.552. Theoretically, for enough large N, the average Manhattan distance of all pairs of two nodes in a $\sqrt{N} \times \sqrt{N}$ grid graph can be evaluated as follows:

$$\iiiint_D (|x-x'|+|y-y'|)dxdydx'dy' = \frac{2}{3}\sqrt{N},$$

where the domain D is $\{0 \le x,y,x',y' \le \sqrt{N}\}$. Similarly, that of a $\sqrt{\frac{N}{2}} \times \sqrt{2N}$ diagrid graph arranged in a square field of size $\sqrt{N} \times \sqrt{N}$ is computed as follows.

$$\iiiint_D \sqrt{2} \cdot \max(|x-x'|,|y-y'|)dxdydx'dy' = \frac{7\sqrt{2}}{15}\sqrt{N}.$$

TABLE III. THE VALUES OF m, $d_{0,0}$, AND $md_{0,0}$ FOR A 4-REGULAR 3-RESTRICTED DIAGRID GRAPH OF SIZE 10×10

i	0	1	2	3	4	5
$m(i)$	1	5	17	53	98	98
$d_{0,0}(i)$	1	8	25	50	85	98
$md_{0,0}(i)$	1	5	17	50	85	98

From $\frac{2}{3} \approx 0.667$ and $\frac{7\sqrt{2}}{15} \approx 0.660$, the average distances of nodes are almost the same, and so the ASPL will be also.

Next, let us discuss the lower bound of the ASPL. As before, let $m(i)$ be the number of nodes in i hops from a particular node of a K-regular graph. Also, let $d_{x,y}(i)$ be the number of nodes in i hops from node (x,y) of an L-restricted diagrid graph. Further, let $md_{x,y}$ be the minimum of them. Table III shows these values of a 4-regular 3-restricted diagrid graph. Using the values of $md_{x,y}(i)$ we can compute the ASPL lower bound A^- of a diagrid graph. For example, $A^- = 3.279$ for a 4-regular 3-restricted diagrid graph. Hence, the gap of the randomly optimized diagrid graph in Figure 7 is $\frac{3.459-3.279}{3.279} = 5.5\%$. Also, from Table III, we have the lower bound $D^- = 5$ of the diameter. Thus, the randomly optimized diagrid graph is diameter-optimal.

Let us compare the diameters of grid graphs and diagrid graphs with almost the same nodes. We use 900-node grid graphs of size 30×30 and 882-node diagrid graphs of size 21×42. Figure 8 shows the diameter of the grid graph and the diagrid graph for $K = 3$, 5, and 10. When L is small, say $L = 2$, the diameter of the grid graphs is 29, while that of the diagrid graphs is 21 for all K. Hence, the diameter is decreased to $\frac{21}{29} = 72.4\%$, which is close to 70.7% obtained by the theoretical analysis. On the other hand, for large L, the diameter is determined by the value of K. For example, $D^+(3,16) = 11$, $D^+(5,16) = 6$, and $D^+(10,16) = 4$ for both the grid graphs and the diagrid graphs. Hence, we should select the diagrid graph if L is not so large.

Finally, we will show that the ASPLs of the grid graph and the diagrid graph are almost the same. Since the average distance of nodes are almost the same, we can expect that the ASPLs are also the same. We can confirm this fact by experiments. Figure 9 shows ASPL $A^+(K,L)$ of grid graphs and diagrid graphs for $K = 3$, 5, and 10. We can see that the

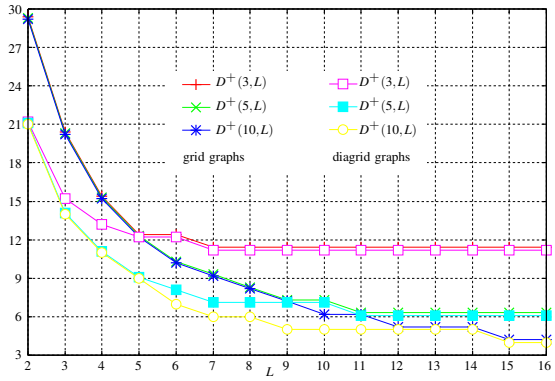


Fig. 8. The diameter $D^+(K,L)$ of grid graphs and diagrid graphs for K =3, 5, and 10

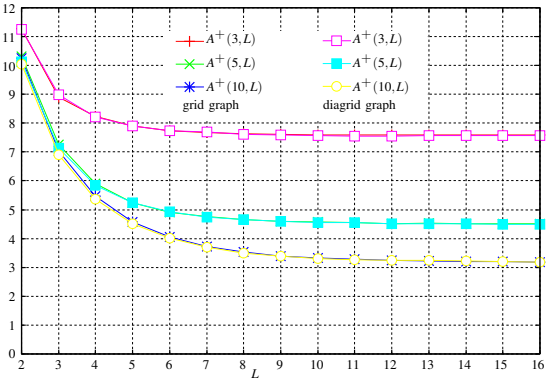


Fig. 9. **ASPL** $A^+(K,L)$ of grid graphs and diagrid graphs for $K = 3$, 5, and 10

ASPL is almost the same for every pair of K and L.

VII. A Guideline for Selecting Degree K and Maximum Length L

As we can see in Section V, we should choose well-balanced values of degree K and maximum length L. Clearly, since the values of K and L determine the hardware resources, smaller values are better to reduce the installation and running costs. Because the upper bound and the lower bound of the ASPL of a K-regular L-restricted grid graph are very close, we use the lower bounds to discuss appropriate selection of values of K and L. The discussion in this section can be applied to diagrid graphs as it is.

Imbalanced values of K and L waste the hardware resources. For example, when $N = 30 \times 30$, $K = 4$, and $L = 8$, $A^-(4,8) = 5.207$, that is, the lower bound of the ASPL is 5.207. Also, $A_m^-(4) = 5.204$ and $A_d^-(8) = 2.939$, that is, the ASPL lower bounds of 4-regular and 8-restricted grid graphs are 5.204 and 2.939, respectively. Hence, the ASPL is determined almost only by K and the maximum length L affects the ASPL a few. Even if we decrease L by one, the ASPL $A^-(4,7) = 5.225$ is almost the same. This fact means that K is too small or L is too large. Such imbalanced values of K and L waste the cost incurred by large L. We should choose smaller L or larger K to balance the contribution of K and L for the ASPL. From this observation, we say that the values of K and L are *well-balanced* if the absolute difference

TABLE IV. Well-balanced pairs of K and L with the lower bounds $A^-m(N,K)$, $A_d^-(K)$ and $A^-(K,L)$ for $N = 30 \times 30$

K	3	4	5	6	9	10
L	3	4	5	6	7	8
$A_m^-(K)$	7.325	5.204	4.377	3.746	3.169	2.877
$A_d^-(L)$	7.000	5.376	4.440	3.751	3.287	2.939
$A^-(K,L)$	8.112	6.001	4.957	4.305	3.626	2.964

$|A_m^-(K) - A_l^-(L)|$ is a local minimum, that is, it is no larger than the absolute differences of four neighbors $|A_m^-(K-1) - A_l^-(L)|$, $|A_m^-(K+1) - A_l^-(N,L)|$, $|A_m^-(K) - A_l^-(L-1)|$, and $|A_m^-(K) - A_l^-(L+1)|$.

Table IV shows a list of well-balanced pairs for a grid graph of size $N = 30 \times 30$. It is recommended to use these pairs when a network of size 30×30 is installed. However, it is not a must to select a pair in this table. We can select a pair close to these well-balanced values. For example, if we select $K = 4$ and $L = 5$, the lower bounds are $A_m^-(4) = 5.204$, $A_d^-(5) = 4.401$, and $A^-(4,5) = 5.471$. Hence, this pair is still acceptable for network installation.

Next, let us discuss asymptotic analysis of well-balanced pairs of K and L of a $\sqrt{N} \times \sqrt{N}$ grid graph from a theoretical point of view. To simplify the analysis, we use big-theta Θ such that $f(n) = \Theta(g(n))$ if $c_1 \cdot g(n) \leq f(n) \leq c_2 \cdot g(n)$ for some c_1 and c_2. From Formulae (1) and (2), we have $A_m^-(N,K) = \Theta(\log N / \log K)$. From Formulae (3) and (4), we also have $A_d^-(N,K) = \Theta(\sqrt{N}/L)$. To balance K and L, we must satisfy $A_m^-(N,K) \approx A_d^-(N,K)$, that is,

$$\Theta(\log N / \log K) \quad \approx \quad \Theta(\sqrt{N}/L). \tag{5}$$

We show several interesting observations from Formula (5). Since we have three parameters N, K, and L, we fix one of them and see the relation of the remaining two parameters. In each observations, we first assume that we have an optimized grid graph of size $\sqrt{N} \times \sqrt{N}$ with parameters K and L.

(1) N is fixed: If L is doubled, then K must be squared from $\log K^2 = 2 \log K$ to keep well-balanced. This means that increment of L affects the ASPL more than that of K. For example, balanced pairs (K,L) for a grid graph of size 300×300 include $(6,32)$ and $(33,64)$, which approximate this relation.

(2) K is fixed: From Formula (5), we have an increasing function $L = \Theta(\log K \sqrt{N} / \log N)$ of N. If N is increased by a factor of α, L must be about $\sqrt{\alpha}$ times larger to keep well-balanced. For example, if $N = 10 \times 10$, then $(K,L) = (6,3)$ is well-balanced. We have shown $(K,L) = (6,6)$ is well-balanced when $N = 30 \times 30$. Since $\alpha = 3$, the increasing ratio $6/3 = 2$ is close to $\sqrt{\alpha} = 1.732$.

(3) L is fixed: From Formula (5), we have a decreasing function $\log K = \Theta(L \log N / \sqrt{N})$ of N. Thus, if L is fixed and N is increased, K must be decreased to keep well-balanced. For example, if $N = 20 \times 20$, then $(K,L) = (11,6)$ is well-balanced. Also, $(K,L) = (6,6)$ is well-balanced when $N = 30 \times 30$. Hence, by increasing the number of nodes, the value of K is decreased to keep well-balanced. Quite surprisingly, the relation of N and K for a fixed L is against this intuition. For example, suppose that a computer manufacturer releases two supercomputers: the mid-range with 20×20 nodes and the high-end with 30×30 nodes. We assume that electric communication cables with the same technology are used

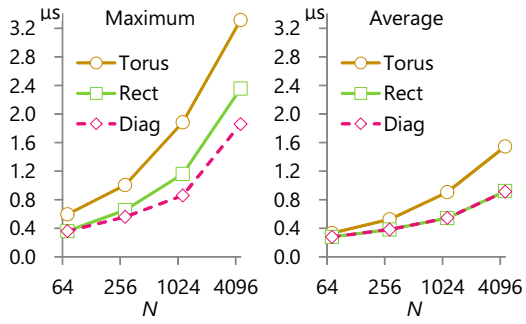


Fig. 10. Zero-load latency for each topology

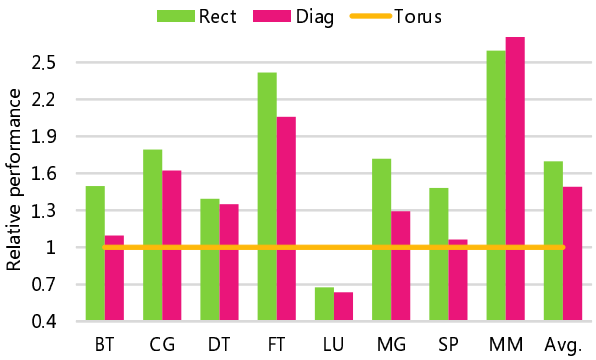


Fig. 11. Application performance for each topology

in these supercomputers and their length must be no more than 6. Manufacturer may want to use more ports in each node of the high-end supercomputer to differentiate the two models. However, the high-end supercomputer should have fewer ports to keep well-balanced. Each node of a 20×20-node supercomputer should have 11 ports, while 6 ports are sufficient for a 30×30-node supercomputer. This is a very interesting observation against developer's common intuition.

VIII. Case Studies

A. Design of Off-chip Low-latency Networks

In the first case study we aim at building low-latency interconnection networks without active optical cables, because they are quite expensive when compared to other network components, switch chips, network interface and passive electric cables[15]. To this end our K-regular L-restricted graph generation method is a perfect way, because the passive electric cables have a limited maximum length.

1) Condition: We assume N switches enclosed in N cabinets arranged on a machine room floor. We set the switch delay to 60 ns and the cable delay to 5 ns/m. The network topologies are generated as in Sections III and VI with parameters $K = 6$ and $L = 6$ so as to use no optical cables. The objective is to minimize the diameter and the ASPL. As a competitor topology we choose a k-ary 3-cube (i.e., a 3-D torus) because other popular topologies (fat tree, Dragonfly, etc.) are difficult to build without long cables. For simplicity we set the cabinet size to 1×1 m on the floor, while a different size is used in the following case studies.

2) Average and Worst Zero-load Latency: Figure 10 illustrates the zero-load communication latency calculated as the sum of the switch delay and the cable delay for the shortest paths (assuming a minimal routing) between every pair of switches. Our randomly optimized grid and diagrid topologies are denoted as Rect and Diag, respectively.

For the 4,608-switch networks, the average zero-load latencies are 921 ns and 915 ns for the grid and the diagrid, respectively. They are about 41% lower than that of the counterpart 3-D torus. The difference in the average latency between the grid and the diagrid is negligible. The maximum (worst) zero-load latencies are 2,355 ns and 1,860 ns for the grid and the diagrid, respectively. The latter is 44% lower than that of the torus. All those results indicate that both the grid and the diagrid offer lower latencies than torus regardless of the network size, and the diagrid has a notable advantage over the grid in the maximum latency.

3) Event-discrete Simulation: We use the SimGrid simulation framework (v3.12)[16]. SimGrid implements validated simulation models, is scalable, and makes it possible to simulate the execution of unmodified parallel applications that use the Message Passing Interface (MPI). We simulate the execution of the NAS Parallel Benchmarks (version 3.3.1, MPI versions)[17] (Class B) and the matrix multiplication example provided in the SimGrid distribution (MM). In all figures the execution times relative to that of the execution on 3-D torus are shown. The higher values are better. We pick 288-switch networks for the fair comparison of the grid and the diagrid. The cable length between switches is set to 5m for all the topologies. We configure SimGrid to utilize its built-in version of the MVAPICH2 implementation of MPI collective communications.

Figure 11 shows the performance of each benchmark executed on our randomly optimized grid (Rect) and diagrid (Diag) topologies, both normalized to that executed on torus. The grid and the diagrid outperform torus by 70% and 49% on average, respectively. CG and LU typically communicates between neighboring switches (i.e., stencil communication), whereas FT, IS, and MM communicates between all pairs of switches (i.e., all-to-all communication). Our randomly optimized topologies achieve higher performance for FT and MM, because they have lower zero-load communication latencies than torus. As shown in Section VIII-A2, the maximum zero-load latency of the diagrid is lower than that of the grid, while the averages are almost the same. As a result, the diagrid slightly outperforms the grid. We thus consider that the maximum zero-load latency affects the all-to-all communication performance.

B. Design of Off-chip Low-power Networks under 1μs Maximum Communication Latency

The second case study aims at generating the lowest-power interconnection networks that satisfies 1μs maximum zero-load latency. The low power consumption is a general requirement and the 1μs latency is listed in the requirements of future interconnection networks[6]. In this case study, we can use both passive electric and active optical cables, but try to minimize the number of active optical cables so as to minimize the power consumption. In this evaluation we do

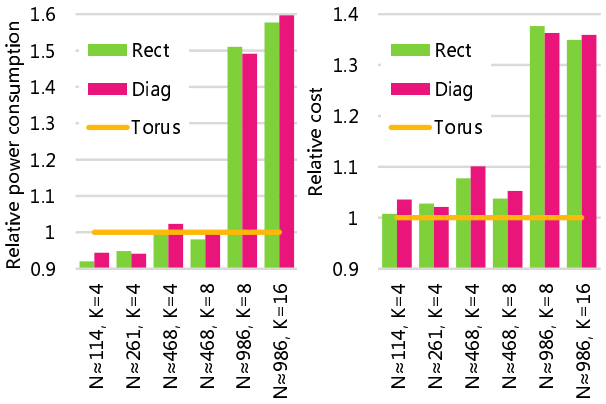
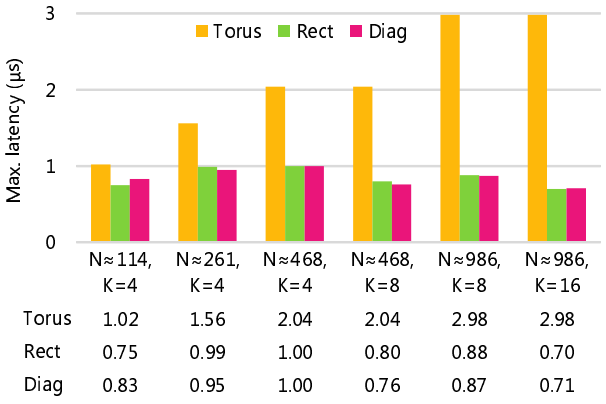


Fig. 12. Network power consumption (left) and cost (right) for each topology.



	N≈114, K=4	N≈261, K=4	N≈468, K=4	N≈468, K=8	N≈986, K=8	N≈986, K=16
Torus	1.02	1.56	2.04	2.04	2.98	2.98
Rect	0.75	0.99	1.00	0.80	0.88	0.70
Diag	0.83	0.95	1.00	0.76	0.87	0.71

Fig. 13. Maximum zero-load latency after optimization.

not consider any link regulation mechanism such as Energy Efficient Ethernet (EEE), with which the NIC/switch PHY is set to Low Power Idle mode when there is no traffic on a cable. The reason is that these operation overheads degrade the performance of latency-sensitive HPC benchmarks, and hence the EEE is not common in the HPC field.

1) Condition: We take the evaluation parameters from the Mellanox Technologies products. The passive electric cable length is assumed to be up to 7 m according to the 40 Gbps InfiniBand cable products. The power consumption of a switch is minimally set to 111.54 W when connected only to passive electric cables, and maximally 200.4 W when connected only to active optical cables[18]. We assume that each cabinet uses the space of 0.6 $\times$ 2.1 m and requires 1 m overhead at both ends of a cable. The other parameters are the same as those in the previous subsection. We use the randomized algorithm for generating a K-regular L-restricted grid graph for the purpose. The 2-opt operation in Step 3 in Section III is implemented as follows. (1) Swap the endpoints of two disjoint edges if the maximum zero-load communication latency becomes lower in this operation. Repeat the procedure until the maximum zero-load communication latency becomes lower than 1 µs. (2) Repeatedly swap the endpoints of two disjoint edges only when (a) the maximum zero-load communication latency is lower than 1 µs and (b) the amount of power consumption for the network decreases in this operation.

2) Network Power Consumption and Cost: Figure 12 (left) shows the network power consumption of the grid (Rect) and the diagrid (Diag) topologies, normalized to the performance of the counterpart torus. The maximum zero-load communication latency after the optimization are shown in Figure 13. Most cases for torus cannot meet the latency requirement. By contrast, most cases for the grid and the diagrid meet, though they increase the network power consumption. Interestingly, the ratio of electric cables over the total number of inter-switch cables varies from 19% to 100% in the grid and the diagrid. Our finding is that the grid and the diagrid topologies work well to the purposes of reducing both the communication latency and the power consumption.

The cable media also affects the network cost. Figure 12 (right) computes the cost for each network topology using the cost model of InfiniBand QDR electric and optical cables[19]. We can see that the cost of the diagrid and the grid increases by 0.7%–33% when compared to that of torus, which does not meet the 1 µs maximum zero-load communication latency.

C. Design of Low-latency On-chip Networks

The last case study focuses on on-chip interconnection networks, especially those used for chip multi-processors (CMPs), in which communication latency significantly affects the parallel application performance.

As an alternative to regular topologies where the longest wire length is automatically determined, the proposed low-latency topology optimized for a given longest wire length parameter would become an attractive design option when the wire delay is a severe design issue. Due to page limitation, the primary objective of this case study is to demonstrate potentials of the optimized low-latency topologies when they are applied for on-chip interconnection networks. To this end, the following three 72-node topologies are compared in terms of average hop count, network latency, and application execution time; 9×8 2-D folded torus (Torus), 9×8 randomly optimized grid topology (Rect), and 12×6 randomly optimized diagrid topology (Diag). The grid and the diagrid are generated by the proposed method under constraints where $K = 4$ and $L = 4$, in order to see a performance gain when L constraint is relaxed compared to the baseline torus. Although manufacturability, yield, and reliability aspects of a diagonal VLSI architecture have been addressed[20], we mainly compare torus and the grid in this work. XY dimention-order minimum routing is used for torus, while a deterministic routing restricted by Up*/Down* rule is used for the grid and the diagrid.

We assume shared-memory CMPs, in which each processor has private L1 data and instruction caches, while the unified L2 cache banks are shared by all the processors. Eight processors (CPUs), 64 L2 cache banks, and four memory controllers are interconnected by 72-node torus, the grid, and the diagrid topologies. CPUs are connected to routers on chip edges (two CPUs for each edge). We used a full-system CMP simulator gem5[21]. The processor and network parameters are listed in Table V. We use eight parallel programs from the OpenMP implementation of NAS Parallel Benchmarks (NPB). The number of threads was set to eight as the number of processors in the target CMPs is eight. Figure 14 shows the application execution times. The results are normalized so that the execution time on torus is 100%. As expected, the application

TABLE V. SIMULATION PARAMETERS OF TARGET CMPS.

Processor architecture	x86-64
L1 I/D cache size, latency	32 KB (line:64B), 1 cycle
L2 cache bank size, latency	256 KB (assoc:4), 6 cycles
Memory size, latency	4 GB, 160 cycles
Cache coherency protocol	MOESI directory
Router pipeline	3 cycles for router + 1 cycle for link
Buffer size, flit size	5 flits per VC, 128 bits
Packet size	1 flit for control, 5 flits for data

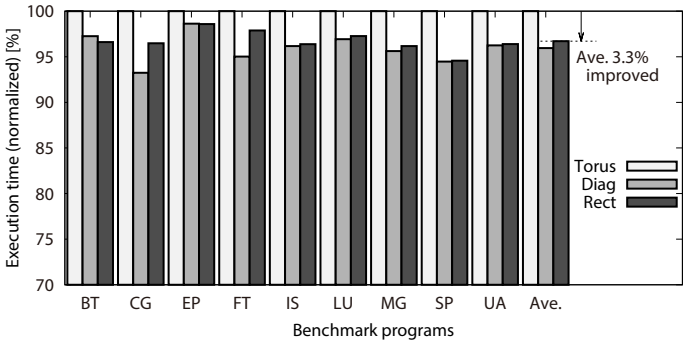


Fig. 14. Full-system simulation results.

execution time results also reflect the communication latency reduction.

IX. CONCLUSIONS

We generate a randomly optimized grid graph so as to satisfy given degree and edge length constraints. Its diameter and ASPL are close to the lower bounds. In addition, we present a diagrid graph to reduce the maximum Manhattan distance of a link, and the diagrid successfully reduces the distance by 29.3%.

Our result graphs can be applied for network topology of an interconnection network. Through three case studies, our main finding is that our randomly optimized topologies reduce the zero-load communication latency, thus giving the performance improvement of seven out of eight parallel applications (70% improvement on average) on off-chip networks. Another finding is that our randomly optimized topologies provide good design trade-off between power consumption, cost, and communication latency. Similarly, they can be applied to an on-chip network. The full-system simulation results illustrate that all the application performance is improved by 3.3% on average. Through these theoretical and practical findings, our recommendation is to use the randomly optimized topologies for various off- and on-chip interconnection networks.

ACKNOWLEDGMENTS

This work was partially supported by MIC/SCOPE (152103004), JSPS KAKENHI (15K00144, 16H02807) and JST CREST.

REFERENCES

[1] "Graph Golf." http://research.nii.ac.jp/graphgolf/.

[2] Combinatorics Wiki, "The Degree Diameter Problem for General Graphs." http://combinatoricswiki.org/wiki/The_Degree_Diameter_Problem_for_General_Graphs.

[3] B. Towles, J. P. Grossman, B. Greskamp, and D. E. Shaw, "Unifying on-chip and inter-node switching within the Anton 2 network," in *Proc. of the ACM/IEEE International Symposium on Computer Architecture, (ISCA)*, pp. 1–12, June. 2014.

[4] Y. Ajima, S. Sumimoto, and T. Shimizu, "Tofu: A 6D Mesh/Torus Interconnect for Exascale Computers," *IEEE Computer*, vol. 42, pp. 36–40, 2009.

[5] V. G. Cerf, D. D. Cowan, R. C. Mullin, and R. G. Stanton, "A lower bound on the average shortest path length in regular graphs," *Networks*, vol. 4, no. 4, pp. 335–342, 1974.

[6] K. S. Hemmert, J. S. Vetter, K. Bergman, C. Das, A. Emami, C. Janssen, D. Panda, C. Stunkel, K. Underwood, and S. Yalamanchili, "Report on Institute for Advanced Architectures and Algorithms, Interconnection Networks Workshop 2008." http://ft.ornl.gov/pubs-archive/iaa-ic-2008-workshop-report-final.pdf.

[7] M. Koibuchi, H. Matsutani, H. Amano, D. F. Hsu, and H. Casanova, "A Case for Random Shortcut Topologies for HPC Interconnects," in *Proc. of the International Symposium on Computer Architecture (ISCA)*, pp. 177–188, 2012.

[8] J. Flich, T. Skeie, A. Mejia, O. Lysne, P. Lopez, A. Robles, J. Duato, M. Koibuchi, T. Rokicki, and J. C. Sancho, "A Survey and Evaluation of Topology Agnostic Deterministic Routing Algorithms," *IEEE Trans. on Parallel Distrib. Syst.*, vol. 23, no. 3, pp. 405–425, 2012.

[9] I. Fujiwara, M. Koibuchi, H. Matsutani, and H. Casanova, "Swap-and-randomize: A method for building low-latency HPC interconnects," *IEEE Trans. Parallel Distrib. Syst.*, vol. 26, no. 7, pp. 2051–2060, 2015.

[10] J. Kim, J. Balfour, and W. J. Dally, "Flattened Butterfly Topology for On-Chip Networks," in *Proceedings of the International Symposium on Microarchitecture (MICRO'07)*, pp. 172–182, Dec. 2007.

[11] J. D. Balfour and W. J. Dally, "Design Tradeoffs for Tiled CMP On-Chip Networks," in *Proceedings of the International Conference on Supercomputing (ICS'06)*, pp. 187–198, 2006.

[12] H. Yang, J. Tripathi, N. D. E. Jerger, and D. Gibson, "Dodec: Random-link, low-radix on-chip networks," in *47th Annual IEEE/ACM International Symposium on Microarchitecture, MICRO 2014, Cambridge, United Kingdom, December 13-17, 2014*, pp. 496–508, 2014.

[13] T. Krishna, C. O. Chen, S. Park, W. Kwon, S. Subramanian, A. P. Chandrakasan, and L. Peh, "Single-cycle multihop asynchronous repeated traversal: A SMART future for reconfigurable on-chip networks," *IEEE Computer*, vol. 46, no. 10, pp. 48–55, 2013.

[14] G. Michelogiannakis, J. D. Balfour, and W. J. Dally, "Elastic-buffer flow control for on-chip networks," in *15th International Conference on High-Performance Computer Architecture (HPCA-15 2009), 14-18 February 2009, Raleigh, North Carolina, USA*, pp. 151–162, 2009.

[15] J. Kim, W. J. Dally, S. Scott, and D. Abts, "Technology-Driven, Highly-Scalable Dragonfly Topology," in *Proc. of the International Symposium on Computer Architecture (ISCA)*, pp. 77–88, 2008.

[16] H. Casanova, A. Giersch, A. Legrand, M. Quinson, and F. Suter, "Versatile, Scalable, and Accurate Simulation of Distributed Applications and Platforms," *Journal of Parallel and Distributed Computing*, vol. 74, no. 10, pp. 2899–2917, 2014.

[17] The NAS Parallel Benchmarks. http://www.nas.nasa.gov/Software/NPB/.

[18] "IS5024, Mellanox Technologies." http://www.mellanox.com/related-docs/user_manuals/IS5024_User_Manual.pdf.

[19] M. Besta and T. Hoefler, "Slim Fly: A Cost Effective Low-diameter Network Topology," in *Proc. of the International Conference for High Performance Computing, Networking, Storage and Analysis*, pp. 348–359, 2014.

[20] N. D. Arora, L. Song, S. M. Shah, K. Joshi, K. Thumaty, A. Fujimura, L. C. Yeh, and P. Yang, "Interconnect Characterization of X Architecture Diagonal Lines for VLSI Design," *IEEE Transactions on Semiconductor Manufacturing*, vol. 18, pp. 262–271, May 2005.

[21] N. L. Binkert, B. M. Beckmann, G. Black, S. K. Reinhardt, A. G. Saidi, A. Basu, J. Hestness, D. Hower, T. Krishna, S. Sardashti, R. Sen, K. Sewell, M. Shoaib, N. Vaish, M. D. Hill, and D. A. Wood, "The gem5 simulator," *SIGARCH Computer Architecture News*, vol. 39, no. 2, pp. 1–7, 2011.