



PolarStar: Expanding the Horizon of Diameter-3 Networks

Kartik Lakhotia
kartik.lakhotia@intel.com
Intel
Santa Clara, CA, USA

Laura Monroe
lmonroe@lanl.gov
Los Alamos National
Laboratory
Los Alamos, NM, USA

Kelly Isham
kisham@colgate.edu
Colgate University
Hamilton, NY, USA

Maciej Besta
mbesta@inf.ethz.ch
ETH Zürich
Zurich, Switzerland

Nils Blach
nils.blach@inf.ethz.ch
ETH Zürich
Zurich, Switzerland

Torsten Hoefler
torsten.hoefler@inf.ethz.ch
ETH Zürich
Zurich, Switzerland

Fabrizio Petrini
fabrizio.petrini@intel.com
Intel
Santa Clara, CA, USA

ABSTRACT

We present PolarStar, a novel family of diameter-3 network topologies derived from the star product of low-diameter factor graphs.

PolarStar gives the largest known diameter-3 network topologies for almost all radices, thus providing the best known scalable diameter-3 network. Compared to current state-of-the-art diameter-3 networks, PolarStar achieves $1.3\times$ geometric mean increase in scale over Bundlefly, $1.9\times$ over Dragonfly, and $6.7\times$ over 3-D HyperX. PolarStar has many other desirable properties, including a modular layout, large bisection, high resilience to link failures and a large number of feasible configurations for every radix.

We give a detailed evaluation with simulations of synthetic and real-world traffic patterns and show that PolarStar exhibits comparable or better performance than current diameter-3 networks.

CCS CONCEPTS

• Networks → Network architectures; Network performance evaluation; Data center networks; • Mathematics of computing → Graph theory; Extremal graph theory.

KEYWORDS

Networks; High-Performance Networks; Network Topology; Network Architecture; Network Evaluation; Extremal Graph Theory

This research is, in part, based upon work supported by the Office of the Director of National Intelligence (ODNI), Intelligence Advanced Research Projects Activity (IARPA), through the Advanced Graphical Intelligence Logical Computing Environment (AGILE) research program, under Army Research Office (ARO) contract number W911NF22C0081. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of the ODNI, IARPA, ARO, or the U.S. Government. This work was also supported by the U.S. Department of Energy through the Los Alamos National Laboratory, operated by Triad National Security, LLC, for the National Nuclear Security Administration of U.S. Department of Energy (Contract No. 89233218CNA000001), and by the U.S. DOE LDRD program at Los Alamos National Laboratory under project number 20230692ER. The U.S. Government retains an irrevocable, nonexclusive, royalty-free license to publish, translate, reproduce, use, or dispose of the published form of the work and to authorize others to do the same. This paper has been assigned the LANL identification number LA-UR-22-30347, ver. 2.



This work is licensed under a Creative Commons Attribution International 4.0 License.

SPAA '24, June 17–21, 2024, Nantes, France
© 2024 Copyright held by the owner/author(s).
ACM ISBN 979-8-4007-0416-1/24/06.
<https://doi.org/10.1145/3626183.3659975>

ACM Reference Format:

Kartik Lakhotia, Laura Monroe, Kelly Isham, Maciej Besta, Nils Blach, Torsten Hoefler, and Fabrizio Petrini. 2024. PolarStar: Expanding the Horizon of Diameter-3 Networks. In *Proceedings of the 36th ACM Symposium on Parallelism in Algorithms and Architectures (SPAA '24)*, June 17–21, 2024, Nantes, France. ACM, New York, NY, USA, 13 pages. <https://doi.org/10.1145/3626183.3659975>

1 INTRODUCTION

1.1 Motivation

The largest current supercomputers contain tens to hundreds of thousands of processing nodes. For example, Frontier, the most powerful system in the Top 500 list [49], has 9, 408 CPUs and 37, 632 GPUs [13]. The fourth-ranked supercomputer, Fugaku, has 158, 976 processing nodes [20]. Future supercomputers will be even larger, since the size of the system determines peak compute performance.

Equally important is the diameter, as this affects communication latency and injection bandwidth per switch. Low-diameter networks, especially of diameters 2 [33] and 3 [36], are of great interest, providing low-latency and cost-effective high-bandwidth communication infrastructure.

We address here this question: what is the largest diameter-3 network topology that can be built using switches of a given radix?

Current technological advances make this question especially timely. The emergence of high-radix optical IO modules with high shoreline density has increased interest in scalable low-diameter networks [17, 34, 45, 61]. Co-packaging of these with compute nodes on the same chip greatly enhances available bandwidth per node. Each packet consumes bandwidth on only a few links, limiting negative effects of tail latency and improving system performance [19].

Low-diameter networks are then required to efficiently utilize bandwidth on co-packaged chips. Since each router is integrated with a compute node, scalability of a co-packaged system is identified with the order of the graph defining the network topology.

1.2 Related Work

Given the importance of low-diameter topologies and the technological constraints on switch radix, it makes sense to design networks with the largest order (number of nodes) possible given diameter D and router degree d . This is the *degree-diameter problem* [46], and the order of G is bounded above by the Moore bound [28]. Networks with good *Moore-bound efficiency* (proximity to Moore bound) are not only highly scalable, but also cost-effective and power-efficient

as they can realize a system of given size with relatively lower radix switches and fewer cables. Unfortunately, the largest known graphs for $D > 2$ and $d > 2$ are much smaller in size than the Moore bound.

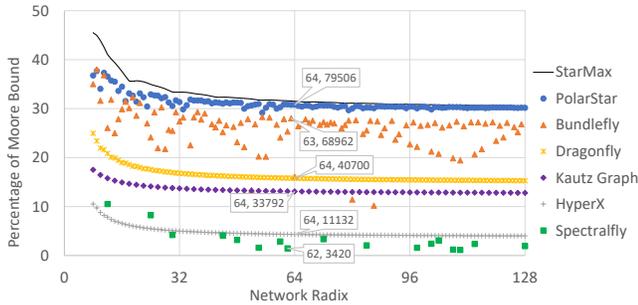


Figure 1: Scalability of direct diameter-3 topologies with respect to the Moore bound. Data labels show the largest number of nodes and corresponding radix in each topology for radix ≤ 64 . *StarMax* denotes an upper bound on the largest graphs theoretically achievable with P - and R -star products – the mathematical constructs used in state-of-the-art Bundlefly and PolarStar networks. For Spectralfly, which is not a fixed diameter topology, we only compare design points with diameter ≤ 3 and largest scale for a given radix (if it exists). For Kautz networks, we consider each link as bidirectional.

Some diameter-2 networks such as Polarfly [37] and Slimfly [7] approach the Moore bound. However, their scale is limited by the small Moore-bound for diameter-2 ($d^2 + 1$ for radix d). Since diameter-2 networks only span a few thousand nodes for feasible radices, they are not suitable for massive-scale datacenters and HPC systems.

On the other hand, diameter-3 networks have a high enough Moore bound to address scalability requirements ($d^3 - d^2 + d + 1$ for radix d). Dragonfly [36] and HyperX [2] are popular diameter-3 topologies deployed in large systems [13, 39], but these topologies exhibit poor Moore-bound efficiency, as shown in Figure 1, which drives up the network cost and power consumption. Kautz networks [44] used in SiCortex’s fabric [59], are directed topologies with near Moore-bound scalability for directed graphs. However, almost all deployed off-chip networking systems [9, 24, 31, 51, 60] and interconnects [2, 35, 36, 42] feature bidirectional links. Treating each link of a Kautz network as bidirectional doubles the degree of nodes, resulting in a higher Moore-bound and lower efficiency. For diameter-3, bidirectional Kautz networks have $< 13\%$ asymptotic Moore-bound efficiency, which is $2.4\times$ smaller than PolarStar.

Recently, Lei et al. introduced Bundlefly [40], a modular diameter-3 network based on star product of two graphs. Its inter-module links can be bundled into multi-core fibers to reduce cabling cost. However, its Moore-bound efficiency varies, as shown in Figure 1.

Creating largest diameter-3 graphs is an open problem in mathematics. There is a big gap between the best-known diameter-3 graphs and the Moore bound. The Combinatorics Wiki leaderboard [46] lists some of the largest diameter-3 graphs as of 2013, up to radix 20. The construction in [6], also utilized by state-of-the-art Bundlefly [40], surpasses the leaderboard for radices 18 – 20. To the best of our knowledge, the construction in [6] also gives the largest known graphs for most radices > 20 , before our construction.

In this paper, we construct a new family of diameter-3 graphs that improve upon the previous best [6, 46] for almost all radices ≥ 18 , achieving a $1.3\times$ geometric mean higher scale than Bundlefly [40].

1.3 Contributions

We propose a new family of network topologies called PolarStar that extends PolarFly [37] to large diameter-3 networks using a mathematical construct called the star product.

- PolarStar gives the *largest known diameter-3 direct networks* for almost all radices, achieving $1.3\times$, $1.9\times$ and $6.7\times$ geometric mean increase in scale over Bundlefly, Dragonfly and HyperX. The base graphs are some of the *largest known diameter-3 graphs*.
- PolarStar reaches *near-optimal scalability* for diameter-3 star products that can be constructed with the currently known properties of the factor graphs that give low-diameter star-products. Further optimizations on these diameter-limited star products are unlikely to provide notable benefits.
- PolarStar *extends several networking benefits of PolarFly*, including a modular layout amenable to bundling of links into multi-core fibers and a large bisection cut.
- PolarStar has a *large design space* and it exists with multiple configurations for every radix in $[8, 128]$.
- We design a *routing mechanism* for PolarStar using its theoretical properties, and show *comparable or better performance than existing diameter-3 networks on synthetic and real-world motifs*.

2 BACKGROUND

2.1 Network Model

In direct networks, each switch is directly linked to or integrated with a compute endpoint. Hence, the topology of a direct network can be modeled as an undirected graph $G(V, E)$ where $V(G)$ is the set of switching nodes, or vertices, $|V(G)|$ is the order of G , and E is the set of links, or edges. Each node has d links to other nodes where d is the *network radix*, or *degree*. The maximum length of shortest paths between any node pair is the *diameter* D . In this paper, we consider networks of diameter 3.

2.2 The Moore Bound

Moore bound [28] is an upper bound on the number of nodes N that a graph of degree d and diameter D may have. This bound is

$$N \leq 1 + d \cdot \sum_{i=0}^{D-1} (d-1)^i.$$

For diameter-3, the Moore bound is $N \leq d^3 - d^2 + d + 1$.

The only graphs with $D \geq 2$ and $d \geq 2$ that achieve the Moore bound are cycles, the Hoffman-Singleton graph, the Petersen graph [5, 15, 28] and a hypothetical diameter-2 degree-57 graph [14]. These graphs are not suited for large-scale network design: degree-2 cycles with low diameter are very small, and the others have only one design point each. Few graphs even come close to the Moore bound. The latest leaderboard of degree-diameter problem from [46] shows that the best known diameter-3 graphs, with the most relevant degrees, reach only 25 – 30% of the Moore bound. The PolarStar construction proposed here is larger than the best known graphs, which are discussed in [6] and [40].

2.3 Network Properties

In this section, we compare PolarStar with Slimfly [7], PolarFly [37], Dragonfly [36], HyperX [2], MegaFly [23, 57], SpectralFly [63], Bundlefly [40] and Fat-trees, on the basis of several desirable network properties. This comparison may be seen in Table 1.

Topology	Direct	Scalability	Stable Design-space	$D \leq 3$	Bundlability
Fat-tree	✘	☐	☐	✘	☐
PolarFly	☐	✘	☐	☐	☐
Slimfly	☐	✘	☐	☐	☐
3-D HyperX	☐	☐	☐	☐	☐
Dragonfly	☐	☐	☐	☐	☐
Bundlefly	☐	☐	☐	☐	☐
MegaFly	✘	☐	☐	☐	☐
Spectralfly	☐	☐	☐	☐	☐
PolarStar	☐	☐	☐	☐	☐

Table 1: Network Properties assessment: battery levels represent network’s standing in terms of the corresponding property “☐”: very good, “☐”: fair, “✘”: not good. D is Network Diameter

Directness: Every switch in a direct network is attached to one or more endpoints, whereas indirect networks have some switching nodes not attached to any endpoint. Directness is a desirable property; for example – if co-packaged modules are used – *indirect* networks such as Fat-tree and MegaFly require fabricating two types of chips, which increases their cost. Further, a switch-only chip in these topologies theoretically requires twice the number of ports than a co-packaged chip with an endpoint.

Scalability: Absolute scale achieved by a topology depends upon the Moore bound for its diameter and its Moore-bound efficiency. Diameter-2 networks such as PolarFly [37] approach the Moore bound but are limited in scale as the bound is small for diameter 2. Three-level Fat-trees scale similarly to diameter-2 networks but additional levels can be used to increase scalability.

Diameter-3 networks have a large Moore bound that can cover hundreds of thousands of nodes with common switch radices. However, known diameter-3 networks such as HyperX and SpectralFly have poor Moore-bound efficiency. PolarStar has the highest Moore-bound efficiency among diameter-3 network topologies.

Low-diameter: Networks with small diameter enable low-latency remote accesses and can sustain high ingestion bandwidth per switch. Diameter ≤ 3 is preferred as it provides both performance and scalability at low cost.

Stable Design-Space: A desirable topology provides several configurations for all radices, and provides stable scaling (consistent Moore-bound efficiency without jitters). This allows system design at various scales (dictated by customer demands) with limited choices for switch radices (constrained by redesign costs, availability etc.). Slimfly and PolarFly [7] have few feasible radices and configurations. Bundlefly’s [40] Moore-bound efficiency fluctuates significantly with the radix (Figure 1). PolarStar features several configurations for every radix and has a more stable Moore-bound efficiency. Although SpectralFly can be constructed for many radices, it has diameter-3 for very few radices, as shown in Figure 1.

Bundlability: Bundlefly [40] introduced the concept of bundling friendly networks as modular topologies with multiple links between adjacent modules (logical groups of nodes) that can be packed together in a Multi-Core Fiber [3]. This reduces cabling cost and complexity. Both Bundlefly and PolarStar are amenable to bundling

as per this description. The largest Dragonfly and MegaFly constructions are less bundling friendly – there is one link between each pair of node groups. Multiple links can be used between adjacent groups to make these topologies bundlable [13]. However, doing so reduces their scalability and Moore-bound efficiency proportionally. Additionally, large racks that can accommodate multiple logical groups can also enable bundling inter-rack cables.

3 APPROACH

The *star product* is a non-commutative product of two factor graphs, presented by Bermond, Delorme and Farhi in [6]. Under certain conditions, the product can achieve a lower diameter than the sum of the diameters of the factors. Lei et al. used a star-product construction for Bundlefly [40].

In this paper, we also use the star product, but devise new properties for our factor graphs that differ from the original ones from [6] used for Bundlefly. These new properties permit larger factor graphs of our own design, and allow us to construct the PolarStar star-product topology, which is larger than Bundlefly and all other previously known diameter-3 networks.

The factor graphs we use not only have large order but also have many other qualities that carry over to the star product, which are useful for networking. We analyze these and the networking characteristics of PolarStar throughout the paper.

4 STAR PRODUCTS

4.1 Intuition for the Star Product

The star product generalizes the Cartesian product $G \times H$. In the Cartesian product, copies of H , called *supernodes*, replace vertices of *structure graph* G . Edges join corresponding vertices in any two copies H_1 and H_2 , when H_1 and H_2 are joined by an edge in G . Figure 2a illustrates this.

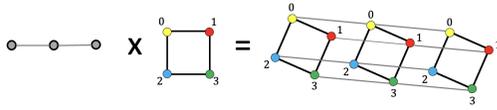
The edges of a star product are also determined by bijections between vertices in neighboring supernodes. In this case, however, the requirement that edges be constructed between corresponding vertices is relaxed. The bijections need not even be the same between different pairs of neighboring supernodes. Carefully selecting the bijective connections may provide a product of diameter at most 1 more than that of the structure graph. We will later discuss how the selection of bijections permits this constrained diameter.

Intuitively, the star product retains the large-scale form of the structure graph G and embeds copies of the supernode H in place of vertices of G . The supernodes are then connected bijectively, with edges as desired. Figure 2 compares a Cartesian product with an example star product on the same factor graphs with its large-scale structure graph and supernode embeddings and connectivity. Figure 5 shows our star product construction.

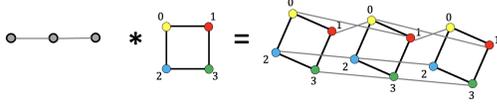
4.2 Formal Definition

The following formal Definition 1 is taken from [6]. In this paper, we call G the *structure graph* and G' the *supernode*.

DEFINITION 1. [6] Let $G = (X, E)$ and $G' = (X', E')$ be two graphs. Take an arbitrary orientation of the edges of G and let U be the resulting set of arcs. For each arc (x, y) in U , let $f_{(x,y)} : X' \rightarrow X'$ be a bijection. The star product $G_* = G * G'$ is defined as follows:



(a) The Cartesian product $L_3 \times C_4$. Corresponding nodes in neighboring copies of C_4 are joined by an edge. The bijection is simply the identity.



(b) An example star product $L_3 * C_4$. In this particular star product, neighboring supernodes are all joined with the same bijection $f = (01)(2)(3)$.

Figure 2: A comparison of the Cartesian product $L_3 \times C_4$ with an example star product $L_3 * C_4$. The structure graph L_3 is the path graph on three vertices, and the supernode C_4 is the cycle graph on four vertices.

- (1) The vertex set of G_* is the Cartesian product $X \times X'$.
- (2) Vertex (x, x') is joined to vertex (y, y') if and only if either
 - (a) $x = y$ and $(x', y') \in E'$, or
 - (b) $(x, y) \in U$ and $y' = f_{(x,y)}(x')$.

In this definition, condition 2a replaces nodes of G with copies of G' . Condition 2b joins copies of G' to each other. The bijection $f_{(x,y)}$ in 2b is the rule for bijective connectivity from vertices of G to vertices of G' across the arc (x, y) .

4.3 Order, Degree, and Diameter

Using the following facts, one may construct a large diameter-constrained graph from two smaller graphs.

- (1) The number of vertices is $|V(G_*)| = |V(G)||V(G')|$.
- (2) If the maximum degrees in G and G' are d and d' respectively, the maximum degree of G_* is $d_* \leq d + d'$.
- (3) If the diameters of G and G' are D and D' , respectively, then the diameter of G_* is given by $D_* \leq D + D'$.

Appropriate bijections for connectivity between adjacent supernodes can reduce the diameter D_* to at most $D + 1$. In this paper, we will use the same bijection f for all arcs of the structure graph, and construct a star product with this restricted diameter. In Section 5.2, we describe such a bijection.

5 A LOW-DIAMETER STAR PRODUCT

Section 4.3 gives upper bounds on degrees and diameters of star products. In Section 5.1, we define properties on the factor graph that allow us to construct a star product with diameter at most 1 more than that of the structure graph.

Bermond et al. describe properties P and P* on graphs G and G' giving large star products with minimal or no increase in diameter over that of G [6]. Our new properties R and R* are similar in spirit but are not the same.

In particular, our R* weakens P* by allowing any diameter, unlike P* which restricts diameter of G' to ≤ 2 . Our R strengthens P by requiring that all vertex pairs be joined by a path of length diameter D and consequently, by a path of length $D + 1$. In contrast, P only requires that vertices with a path of length D between them be joined by a path of length $D + 1$ as well.

Weakening P* permits us to design IQ , a G' supernode graph with Property R* that can have diameter > 2 and a larger order than previously known supernodes with Property P*. Using IQ , and noting that Erdős-Rényi polarity graphs have Property R, we produce a *larger star product* having diameter 3.

Using the new R properties, we thus improve on the two largest known star-product graphs given in [6], with $1.06\times$ and $1.21\times$ geometric mean improvement for radices in the range [8, 128]. In fact, this new family of diameter-3 graphs is *larger than any diameter-3 graphs previously designed* for almost all radices.

More importantly for this community, PolarStar is much larger than state-of-the-art Bundlefly [40], which is a P-star-product based on [6]. Bundlefly does not use our larger supernode, nor does it use an optimal structure graph. PolarStar thus achieves $1.3\times$ geometric mean larger scale than Bundlefly, for network radices in [8, 128].

5.1 Useful Factor-Graph Properties

5.1.1 Structure-Graph Property R. This first property and its associated proposition apply to the structure graph G , and highlight its *path diversity*. We will later see that this diversity enables reachability in $D + 1$ hops between all vertex pairs in the star product.

Property R. A graph G of diameter D has Property R if any vertex pair $x, y \in V(G)$ can be joined by a path of length D .

Note that in the definition of Property R, self-loops (if they exist in G) are permissible as part of the length- D path.

5.1.2 Supernode Properties R* and R₁. The next properties, R* and R₁, apply to the supernode G' . These properties address *paths internal to supernodes*. Given that paths between supernodes are defined by application of a bijection f to supernode vertices, we may use these to define a short path through the star product.

Property R*. Graph G' satisfies Property R* if there is an involution f so that for any $x', y' \in V(G')$, at least one of the following is true:

- (a) $y' = x'$
- (b) $y' = f(x')$
- (c) $(x', y') \in E(G')$
- (d) $(f(x'), f(y')) \in E(G')$

Proposition 1. A graph G' of degree d' having Property R* has at most $2d' + 2$ vertices.

PROOF. Fix some vertex y in G' . We will count how many other vertices there can be. Let x be a vertex in G' . By Property R*, at least one of the following is true:

- (1) $y = x$, giving exactly one choice for x .
- (2) $y = f(x)$: Since f is a bijection, then there is exactly one choice for x , namely $f^{-1}(y)$.
- (3) $(x, y) \in E(G')$, giving $\deg(y)$ choices for x .
- (4) $(f(x), f(y)) \in E(G')$: since f is a function, $f(y)$ is fixed. Therefore there are $\deg(f(y))$ choices for x .

Altogether, we have at most $2 + \deg(y) + \deg(f(y))$ total vertices, assuming all conditions are disjoint. Since $\deg(y) \leq d'$ and $\deg(f(y)) \leq d'$, G' can have at most $2 + 2d'$ vertices. \square

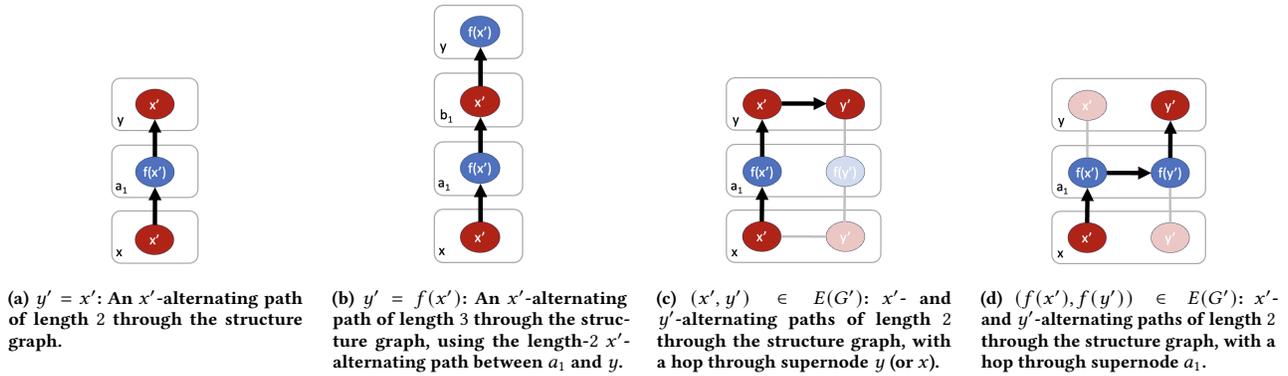


Figure 3: This figure illustrates Theorem 1 on a diameter-2 structure graph with property R , showing how paths are constructed from 2- and 3-hop alternating paths. The star product path may include a 1-hop supernode detour, shown in 3c and 3d, which is permitted by Property R^* on G' .

DEFINITION 2. Let $G * G'$ be a star product where G' has Property R^* . A x' -alternating path in $G * G'$ is a path between supernodes of the form $((a_0, x'), (a_1, f(x')), (a_2, x'), \dots)$, where (a_0, a_1, a_2, \dots) is a path in G , and the second entries in the path elements alternate between x' and $f(x')$.

When G' has Property R^* , the alternating path is the only kind of path that exists between supernodes, by definition of the star product. This is because f is an involution i.e. $f^2(x') = x'$.

Subproperties (c) and (d) of Property R^* allow a hop from any alternating path to any other. This is quite powerful, and is key to Theorem 1, establishing the low diameter of these star products.

LEMMA 1. Let $G * G'$ be a star product where G' has Property R^* . For all $x' \in G'$, there is an alternating path in $G * G'$ for every path in G . All alternating paths are of this form.

PROOF. Follows from the definition of the bijective star product, and the fact that $f^2(x') = x'$ for all x' . \square

The following Property R_1 is Property P_i in [6] for the special case $i = 1$, which is needed for diameter-3 star products.

PROPERTY R_1 . [6] A graph G' has Property R_1 if there is a bijection f , with f^2 an automorphism of G' , so that the set of edges $E(G') \cup f(E(G'))$ is the entire set of edges in the complete graph on $V(G')$.

5.2 Star Products of Low Diameter

Here we give criteria for a star product $G * G'$ to have diameter at most $D + 1$. This occurs when G has diameter D , and either

- The structure graph G has Property R and the supernode G' has Property R^* (Theorem 1), or
- The supernode G' has Property R_1 (Theorem 2, [6]).

We define a path in the star product from (x, x') to (y, y') . The strategy here is to identify x' - and y' -alternating paths of length D from supernodes x (or neighbors of x) to y . We take a single hop from the x' to the y' alternating paths if necessary, using Subproperties (c) and (d) from Property R^* .

Figure 3 illustrates Theorem 1, showing the diameter-3 problem of interest for this paper. The theorem, its proof, and the figure give insight into routing strategies discussed in Section 9.2.

THEOREM 1. Let G and G' be graphs that satisfy Property R and R^* , respectively. Define $f_{(x,y)}(x') = f(x')$ (from Property R^*) for every edge (x, y) of G . If diameter of G is D , the star product $G_* = G * G'$ is a graph with diameter at most $D + 1$.

PROOF. Let (x, x') and (y, y') be nodes in $G * G'$. By Property R of G , there are length- D x' - and y' -alternating paths joining supernodes x and y , and also joining supernodes a and y in $G * G'$, where a is a neighbor of x in G .

We consider the exhaustive set of cases relating x' and y' as described in Property R^* . For simplicity of presentation, we assume that D is even. The same proof holds for D odd, reversing the reasoning for cases (a) and (b).

- (a) If $y' = x'$, we stay in the length- D x' -alternating path.
- (b) If $y' = f(x')$, we take 1 hop to $(a, f(x'))$ (a a neighbor of x in G), then stay in the length- D x' -alternating path from $(a, f(x'))$ to $(y, f(x'))$, giving a length $D + 1$ path.

In the remaining cases, we start with a D -hop x' -alternating path and take 1 hop in some supernode to a parallel D -hop y' -alternating path, giving a length $D+1$ path. Supernode choice depends on which of (x', y') or $(f(x'), f(y'))$ is in $E(G')$.

- (c) If $(x', y') \in E(G')$, we may take 1 hop from (b, x') to (b, y') at any node (b, x') in the x' -alternating path. We reach such (b, x') on even hops.
- (d) If $(f(x'), f(y')) \in E(G')$, we may take 1 hop from $(b, f(x'))$ to $(b, f(y'))$ at any node $(b, f(x'))$ in the x' -alternating path. We reach such $(b, f(x'))$ on odd hops.

\square

Each of the cases in the proof of Theorem 1 are illustrated in Figure 3, for $D = 2$. Figure 3c illustrates case (c), where the intra-supernode hop is taken 2 hops into the path at supernode y . We might also have taken this hop at supernode x which is 0 hops into the path. Likewise, Figure 3d illustrates case (d), where the intra-supernode hop is taken 1 hop into the path at supernode a_1 .

The following theorem from [6] was stated for Property P_i . We restate the theorem for our special case where $i = 1$, to get diameter $D + 1$. The proof is the same as that found in [6].

THEOREM 2. [6] Let G be a graph of diameter $D \geq 2$, and let G' have Property R_1 . Define $f_{(x,y)}(x') = f(x')$ (from Property R_1) for every arc (x, y) of an arbitrary orientation of the edges of G . Then $G * G'$ has diameter at most $D + 1$.

6 CHOOSING GOOD FACTOR GRAPHS

We want to choose the largest feasible structure graph and supernode, since the order of the star product is the product of the orders of the factor graphs. We also want factor graphs with properties useful for networking (including the R properties, among others).

- For the structure graph, we use the *Erdős-Rényi* (ER_q) family of polarity graphs, introduced by Erdős and Rényi [21] and by Brown [11]. This family of graphs has Property R and is larger than any other known family of diameter-2 graphs, so is an excellent structure graph candidate. ER_q has many other networking advantages that the star-product inherits [18, 37, 38].
- For the supernode, we construct a new graph having Property R^* called *Inductive-Quad* (IQ). IQ is larger than all other supernodes with properties known to produce diameter-3 star product with ER_q , and thus, is a better G' candidate than any existing construction. IQ attains the bound on R^* graphs; thus, no supernode with property R^* , P^* or P_1 , can be larger than IQ .

6.1 Choice of the Structure Graph G

6.1.1 ER_q and its Networking Properties. The Erdős-Rényi (ER) or Brown family of polarity graphs [11, 21] is based on finite projective geometry, where adjacency is defined by orthogonality. These graphs were used for the PolarFly [37], SymSig [10] and demi-PN [12] networks, due to their many advantages [37, 38].

These desirable networking properties carry over to the star product. Thus, the family of ER graphs is an excellent candidate for the structure graph, since these graphs do have Property R , as per Theorem 3. Advantages include:

- ER graphs are larger than other known diameter-2 graphs at almost all degrees, as shown in Figure 4. The order of ER graphs for degree $q + 1$ is $q^2 + q + 1$, so they asymptotically reach the diameter-2 Moore bound ($q^2 + 2q + 2$).
- ER has a large number of feasible degrees: an ER graph exists for any degree $q + 1$, where q is a prime power.
- ER has a modular layout.
- ER has high bisection bandwidth.

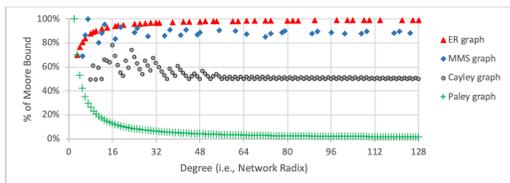


Figure 4: Moore-bound comparison for some known families of diameter-2 graphs: the ER graph, the McKay-Miller-Siráň graphs [48], the best Cayley graphs [1], and the Paley graph. It can be seen that any larger structure graph would only marginally increase the size of the star product.

6.1.2 Construction of ER_q . The vertices of ER_q are vectors (x, y, z) , with $x, y, z \in \mathbb{F}_q$, the finite field of order q . Vertices v and w are adjacent if their dot product $v \cdot w = 0$, with addition and multiplication as in \mathbb{F}_q . Since adjacency is defined by orthogonality of two vectors, all multiples of any two vectors retain the same adjacency relationship. Thus, we move to projective space, considering only the left-normalized form of each vector (so the leftmost non-zero entry of each vector is 1). The ER graph has these left-normalized vectors as the vertices and edges between all orthogonal vector pairs. Note that arithmetic over \mathbb{F}_q is used to compute orthogonality. See [47] for details of arithmetic over \mathbb{F}_q and [37] for ER graph construction.

ER_q is a diameter-2 graph. This may intuitively be seen by considering perpendicularity in Euclidean space. Each pair of distinct vectors v_0 and v_1 is orthogonal to a common $w = v_0 \times v_1$, the cross product of v_0 and v_1 . The 2-hop path from v_0 to v_1 is then given by (v_0, w, v_1) . The intuition is similar in the case of finite geometry.

Certain vertices in ER_q are self-orthogonal, since we use the arithmetic of \mathbb{F}_q . If we allow the self-loops as factor-graph edges, the condition of Property R then holds for all vertices.

THEOREM 3. ER_q has Property R for all prime powers q .

PROOF. Any pair of vertices (x, y) in an ER graph are connected by a 2-hop path via the cross product vertex $w = x \times y$. For some pairs $\{x, y\}$, w may be the same as x or y , in which case the 2-hop path includes a self-loop. \square

Factor graph self-loops then add edges to the star product, as in Figure 5c, and we drop any remaining self-loops in the product.

6.2 Choice of the Supernode G'

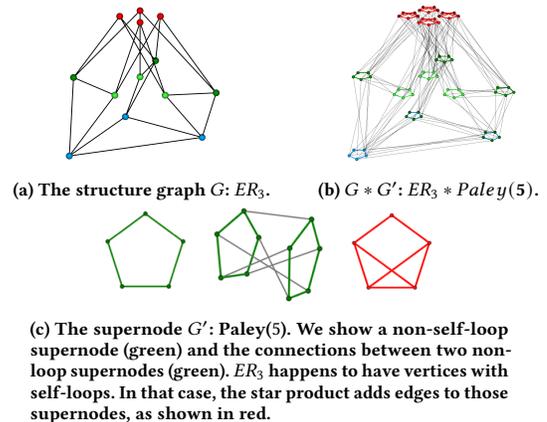


Figure 5: Construction of the star product $ER_3 * Paley(5)$.

For the supernode G' , we construct a new family of graphs, called *Inductive-Quad* (IQ). This family, where it exists, has Property R^* and meets the upper bound on the order of a supernode. We also mention the Paley graph [22, 50] as a G' candidate. It is only slightly smaller, and has symmetry, useful in network design. The Inductive-Quad and Paley graphs give the largest star-products for almost all radices: the IQ supernode almost always produces the largest star product, and Paley produces the largest of the others. We show the star-product construction of PolarStar in Figure 5.

Supernodes	Order	Permitted d'	Symmetric	R^*	R_1
Inductive-Quad	$2d' + 2$	0 or 3 (mod 4)	N	Y	N
Paley [22, 50]	$2d' + 1$	even, $2d' + 1$ a prime power	Y	N	Y
BDF [6]	$2d'$	all	N	Y	N
Cayley [48]	$2d' + \delta, \delta \in \{0, \pm 1\}$	$2d' + \delta$ a prime power	Y	N	Y
Complete	$d' + 1$	all	Y	Y	Y

Table 2: Comparison of parameters of degree d' supernodes.

Other supernode topologies may be of interest, so we mention these here for completeness. For instance, complete graphs provide densely-connected regions of locality, and Cayley graphs are highly symmetric [48]. The BDF graphs are designed in [6] specifically for large star products. We list supernode choices in Table 2, and show the orders of IQ and Paley for relevant radices in Figure 7.

6.2.1 Construction of Inductive-Quad Graphs. We show here a new construction of a degree- d' R^* graph that reaches the upper bound $2d' + 2$ on order of R^* graphs.

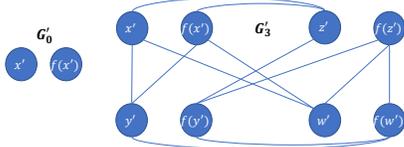
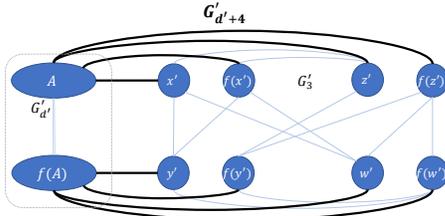
We begin with two base R^* Inductive-Quad graphs shown in Figure 6a: IQ_0 of degree $d' = 0$ with 2 vertices, and IQ_3 of degree $d' = 3$ with 8 vertices. Both have $2d' + 2$ vertices.

We then inductively construct a graph of degree $d' + 4$ with Property R^* from a graph $IQ_{d'}$ of degree d' with Property R^* that has $2d' + 2$ vertices. As shown in Figure 6b, $V(IQ_{d'})$ may be partitioned into two disjoint sets of $d + 1$ vertices each: A and $f(A)$. To construct $IQ_{d'+4}$, we add the 8 vertices of IQ_3 :

$$\{x', f(x'), y', f(y'), z', f(z'), w', f(w')\}$$

along with the IQ_3 edges. Next, we add the following edges:

- between $\{x', f(x'), z', f(z')\}$ and all vertices in A , and
- between $\{y', f(y'), w', f(w')\}$ and all vertices in $f(A)$.

(a) Base Inductive-Quad graphs of degree $d' = 0$ and $d' = 3$.(b) The construction of the Inductive-Quad graph of degree $d' + 4$ from an Inductive-Quad graph of degree d' and IQ_3 .Figure 6: Inductive construction of Inductive-Quad topology with embedded bijection f that satisfies Property R^* .

Proposition 2. $IQ_{d'}$ has $2d' + 2$ vertices and degree $d' = 4n$ or $d' = 4n + 3$, and has Property R^* .

PROOF. The order and degree follow from the construction. Property R^* follows by induction, noting by inspection that IQ_0 and IQ_3

both have Property R^* , and noting that the inductive construction then guarantees Property R^* . \square

Corollary 1. For every integer $n \geq 0$, there exists a graph $G'_{d'}$ of degree $d' = 4n$ or $d' = 4n + 3$ that satisfies R^* and has $2d' + 2$ vertices.

Corollary 1 implies that the IQ graphs are larger than those discussed in [6], as those graphs have a smaller G' supernode.

It is also true that graphs $G'_{d'}$ satisfying R^* and having $2d' + 2$ vertices exist only for $d' = 4n$ or $d' = 4n + 3$ for some non-negative integer n , but we do not prove this here.

7 DESIGN SPACE OF POLARSTAR

We evaluate the scale of network achievable by PolarStar and compare it against existing diameter-3 topologies.

7.1 Theoretical Analysis

Recall the degree of star product $G * G'$ is $\deg(G) + \deg(G')$, and the order is $|V(G)| \cdot |V(G')|$. Our structure graph is ER_q , which has degree $d = q + 1$ and order $q^2 + q + 1$, where q is a prime power. Using a Inductive-Quad supernode of degree d' , we get a PolarStar G_* of degree $d_* = d + d'$ and order

$$|V(G_*)| = (q^2 + q + 1)(2d' + 2) = (q^2 + q + 1)(2d_* - 2q).$$

The order is maximized for

$$\arg \max_q V(G_*) = \frac{(d_* - 1) + \sqrt{(d_* - 1)(d_* - 2)}}{3} \approx \frac{2d_*}{3}. \quad (1)$$

Substituting this value of q , we get that the maximum order of PolarStar for a given degree d_* is

$$\max_{\text{Inductive-Quad}} |V(G_*)| \approx \frac{8d_*^3 + 12d_*^2 + 18d_*}{27}. \quad (2)$$

Similarly, $\max_{\text{Paley}} |V(G_*)| \approx \frac{8d_*^3}{27}$. Thus, PolarStar asymptotically reaches the 8th Moore bound for diameter-3 graphs.

In practice, the degree distribution among factor graphs is constrained as (a) q must be a prime power in an ER graph of degree $q + 1$, and (b) Inductive-Quad and Paley graphs exist for a subset of integer radices, as shown in Table 2. Therefore, we evaluate all feasible combinations of d and $d' = d_* - d$ for both Inductive-Quad and Paley supernodes to find the largest configuration.

7.2 Scalability in Practice

Figure 1 compares the scalability of PolarStar and other direct diameter-3 topologies, in terms of their Moore-bound efficiency. Clearly, PolarStar exceeds the scalability of all known diameter-3 topologies. Compared to HyperX [2] and Dragonfly [36], it achieves 6.7 \times and 1.9 \times geometric mean increase in the order, respectively.

Unlike the state-of-the-art Bundlefly [40], PolarStar offers a construction for every network radix and more consistent scaling with respect to the Moore bound. Overall, it is 1.3 \times geometric mean larger than Bundlefly, which results from the use of more scalable structure graphs and supernodes in its star product, with more possibilities of degree distributions for better optimization of scale.

PolarStar also approaches the optimal scale for a star-product network with the currently known factor graph properties that give a diameter-3 product. This is because (a) the Erdős-Rényi structure

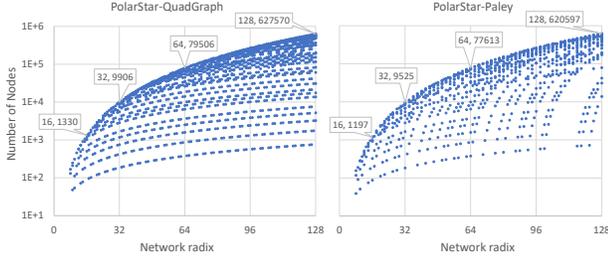


Figure 7: Feasible combinations of radix and order in PolarStar.

graph asymptotically reaches the diameter-2 Moore bound, and (b) the Inductive-Quad supernode topology reaches the optimal order for graphs satisfying Property R*.

For all radixes, the largest PolarStar order for degrees $k \in [8, 128]$ is constructed with the Inductive-Quad supernode, except $k = 23, 50, 56, 80$, where Paley supernode gives a larger topology. PolarStar also offers a wide range of network orders for each radix, as shown in Figure 7. This diversity of feasible designs is enabled by varying (a) the degree distribution between structure and supernode graphs, and (b) the choice of supernode graph.

8 LAYOUT

Physical deployment favors modular topologies comprised of smaller identical subgraphs that may be implemented as blades or racks. Also, if multiple links connect adjacent modules, they can be bundled into Multicore Fibers (MCFs) to reduce cable count [3, 40]. We explain the hierarchically modular structure of PolarStar shown in Figure 8 and point to opportunities for bundling links in PolarStar.

Consider a maximum order PolarStar of degree d_* with ER_q structure graph of degree $q + 1 \approx \frac{2d_*}{3}$ (Equation (1)), and Inductive-Quad supernode of degree $d' = d_* - (q + 1)$. In this PolarStar, the *supernode* is the smallest building block (Figure 5) of $2d_* - 2q$ nodes, and is replicated $q^2 + q + 1$ times (once per node of ER_q). There are $2(d_* - q)$ links between each pair of adjacent supernodes. If these can be bundled in an MCF, we get $q(q + 1)^2$ inter-module MCFs (same as the non-self-loop edges in ER_q [37]). Therefore, such bundling reduces the global cables by a factor of $\approx \frac{2d_*}{3}$.

The next level in modular hierarchy is the *supernode clusters*. As shown in [37], ER_q graph can be divided into $q + 1$ clusters with approximately q links between each pair of clusters. In PolarStar, this translates to $q + 1$ supernode clusters with approximately q bundles of links between each pair as shown in Figure 8c.

9 EVALUATION: SYNTHETIC PATTERNS

9.1 Topologies

We compare PolarStar with Bundlefly [40], Megafly [23, 57] and Spectrallyfly [63] as state-of-the-art diameter-3 networks, 3-D HyperX [2] and Dragonfly [36] as popular diameter-3 networks in practice, and 3-level Fat-trees [43] as the most widely used network, as is standard, using Booksim simulator [30].

Networks such as torus, hypercube or Flattened Butterfly have been shown to have lower performance than these baselines [7, 36]. We also explored the Galaxyfly family of flexible low-diameter

topologies [41]. A diameter-3 Galaxyfly is isomorphic to a Dragonfly, which is included in the comparison.

Network	Parameters	# Routers	Network Radix	# Endpoints
PolarStar with Inductive-Quad (PS-IQ)	$d=12, d'=3, p=5$	1,064	15	5,320
PolarStar with Paley (PS-Pal)	$d=9, d'=6, p=5$	993	15	4,965
Bundlefly (BF)	$d=11, d'=4, p=5$	882	15	4,410
3-D HyperX (HX)	$9 \times 9 \times 8, p=8$	648	23	5,184
Dragonfly (DF)	$a=12, h=6, p=6$	876	17	5,256
Spectrallyfly (SF)	$\rho=23, q=13, p=8$	1,092	24	8,736
Megafly (MF)	$\rho=8, a=16, p=8$	1,040	16	4,160
3-level Fat-tree (FT)	$n=3, p=18$	972	36	5,832

 Table 3: Simulated configurations of topologies. Endpoints per router = p .

The configurations of topologies used are shown in Table 3. For direct diameter-3 networks, p , the number of endpoints per router, is $1/4^{\text{th}}$ of total ports ($1/3^{\text{rd}}$ of the network radix). For Fat-tree and Dragonfly, we use the built-in BookSim constructions and routing [30]. Booksim's Fat-tree topology for router radix $2p$ and 3 layers has $p^2 = 324$ routers in each layer, with top layer routers having half the radix (18). This configuration supports $p^3 = 5,832$ endpoints. In Megafly and Fat-tree, half and one-third of the routers, respectively, have endpoints on half of their ports.

9.2 Minimal Path Computation in PolarStar

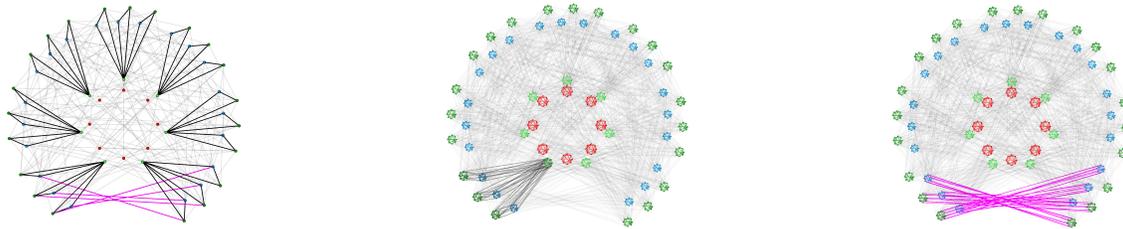
We use the properties of structure graph G and supernode G' to compute each minimal path (minpath) in PolarStar. This significantly reduces the storage requirements of the router compared to table-based routing which stores output port along minpath towards every router in the network.

The structure graph in PolarStar is an *Erdős-Rényi* polarity graph with diameter $D = 2$ and has Property R. We describe minpaths in Inductive-Quad supernode that satisfies Property R*; computation for a Paley supernode is similar. Our minpath computations are amenable to incremental routing and therefore, suitable for *destination-based routing*.

If the source and destination are adjacent, minpath is just the edge between them. Next, we show that when source and destination are in adjacent supernodes, there is a 2-hop minpath between them. These paths are graphically illustrated in Figure 3 when routing from $(a_1, f(x'))$ to (y, y') . For clarity and alignment with Figure 3, we assume $(a_1, f(x'))$ is the source, and (y, y') is the destination. We compute the minimal paths in this case by individually evaluating all scenarios of Property R*:

- (a) $y' = x' \rightarrow (a_1, f(x'))$ and (y, x') are adjacent.
- (b) $y' = f(x') \rightarrow$ We do not hop to supernode y directly from the source. From Property R, there is a 2-hop path (a_1, b_1, y) in structure graph G . We take the corresponding path in G_* which is $((a_1, f(x')), (b_1, x'), (y, f(x')))$, see Figure 3b.
- (c) $(y', x') \in E(G') \rightarrow$ we use the path $((a_1, f(x')), (y, x'), (y, y'))$, as shown in Figure 3c.
- (d) $(f(y'), f(x')) \in E(G') \rightarrow$ we use the path $((a_1, f(x')), (a_1, f(y')), (y, y'))$, as shown in Figure 3d.

Lastly, we consider the case when source and destination are neither adjacent nor in adjacent supernodes. This is also shown in Figure 3 where source is (x, x') and destination is (y, y') . From Property R, there exists a 2-hop path (x, a_1, y) in G . In G_* , we always hop



(a) Modular layout for ER_7 graph [37]. Each group of 3 triangles with a common node is a non-quadric cluster. Red nodes form a quadric cluster. Magenta edges connect two clusters.

(b) Layout of $PolarStar_{11}$ with an ER_7 structure graph. Each ER_7 node becomes an IQ_3 supernode in $PolarStar$. The highlighted links and incident supernodes are a supernode cluster.

(c) Each pair of supernode clusters in $PolarStar$ are connected by multiple link bundles in magenta. Each link bundle corresponds to a single inter-cluster link of ER_7 shown in Figure 8a.

Figure 8: Hierarchical Modular Layout for $PolarStar$ derived from a layout for ER structure graphs used in the $PolarFly$ network [37]. Adjacent supernodes are connected by a bundle of links and adjacent supernode clusters are connected by multiple such bundles.

from (x, x') to $(a_1, f(x'))$. Since $(a_1, f(x'))$ is in a supernode adjacent to (y, y') , we then take the 2-hop path to (y, y') as described above, giving an overall path of length at most 3.

9.3 Routing

We use the following well-known routing schemes:

- *Minimal Routing (MIN)*: Each packet between the source and destination is routed along a shortest path. In MF, we use the path diversity between routers within the same supernode (group). We saw poor performance on SF and BF when using a single minpath per router pair, so for these we store all minpaths for every destination in a large routing table. HX also uses all minpaths but computes them by aligning coordinates in each dimension, so does not require large routing tables [2]. For $PolarStar$, minpath is analytically computed as described in Section 9.2. It only stores shortest paths in the structure graph and requires significantly less memory compared to SF and BF.
- *Load-balancing Adaptive Routing (UGAL)*: Valiant routing balances the load by misrouting each packet via a randomly chosen intermediate router. In our implementation, we sample 4 intermediate routers at random for misrouting. We predict path latency for valiant paths and minpaths using local buffer occupancy, and select the smallest latency path. Note that UGAL routing in SF, BF and HX uses multiple minpaths stored in routing tables.

9.4 Simulation Setup

We analyze network performance using the cycle-accurate BookSim simulator [30]. Simulation parameters (latency, bandwidth) are normalized to the values of a single link. Credit-based flow-control is used to create backpressure and restrict injection for congestion-control. Packets are of size 4 flits and input-queued routers have 128 flit buffers per port and 4 virtual channels (VCs). Dragonfly and Megafly respectively use 2 and 1 VCs for MIN routing, and 3 and 2 VCs for UGAL routing. A warm-up phase precedes simulations so the network may reach steady-state before measurements.

We analyze network performance for synthetic traffic patterns that represent crucial applications. Such patterns are widely used to evaluate network topologies [2, 7, 36, 37, 40].

- (1) *Uniform* random traffic – the destination of each packet is selected uniformly at random (represents graph processing and irregular algorithms e.g. sparse linear algebra [8, 16]).
- (2) *Random permutation* traffic – a fixed permutation mapping τ of source to destination routers is chosen uniformly at random. Endpoints on a router R_s transmit only to corresponding endpoints on router $\tau(R_s)$. This pattern also emulates permutation traffic for co-packaged nodes with both compute and router. Permutation traffic is commonly seen in FFT, physics simulations and collectives [7]. The random permutation pattern represents traffic generated by these applications when process IDs are randomly mapped to nodes.
- (3) *Bit Shuffle* traffic - the destination address is obtained by shifting the source address bits to the left by 1 ($d_i = s_{(i-1) \bmod b}$). This pattern is common in FFT and sorting algorithms [4].
- (4) *Bit Reverse* traffic - the destination address is obtained by reversing the bit order of source address ($d_i = s_{b-i-1}$). This pattern occurs in Cooley-Tukey FFT, binary search and dynamic tree data structures [25, 55, 62].

Bit Shuffle and Bit Reverse traffic uses 2^b endpoints, where 2^b is the largest power of two no more than the total endpoints. Endpoint IDs are contiguous for each router and supernode/group in hierarchical topologies ($PolarStar$, Bundlefly, Dragonfly, Megafly, Fat-tree). In such topologies, almost all endpoints in a supernode communicate with only two other supernodes under Bit Shuffle traffic.

9.5 Results

Figure 9 compares $PolarStar$ performance against the baseline topologies, for different routing schemes and traffic patterns. Labels follow the scheme <topology>-<routing>. Load is normalized to the peak injection bandwidth.

Overall, PS-Pal and PS-IQ perform competitively for most of the patterns. With MIN routing, PS-Pal and PS-IQ sustain more than 75% of full injection bandwidth (load) on uniform traffic. SF and BF use all minpaths available and have marginally better performance, but require significantly higher storage for routing tables than PS-*

With adaptive UGAL routing, PS-Pal and PS-IQ sustain between 0.4 to 0.6 of the full load on various traffic patterns. Their performance is significantly better than DF and is comparable to BF (also a star-product), even though BF stores multiple minpaths for UGAL.

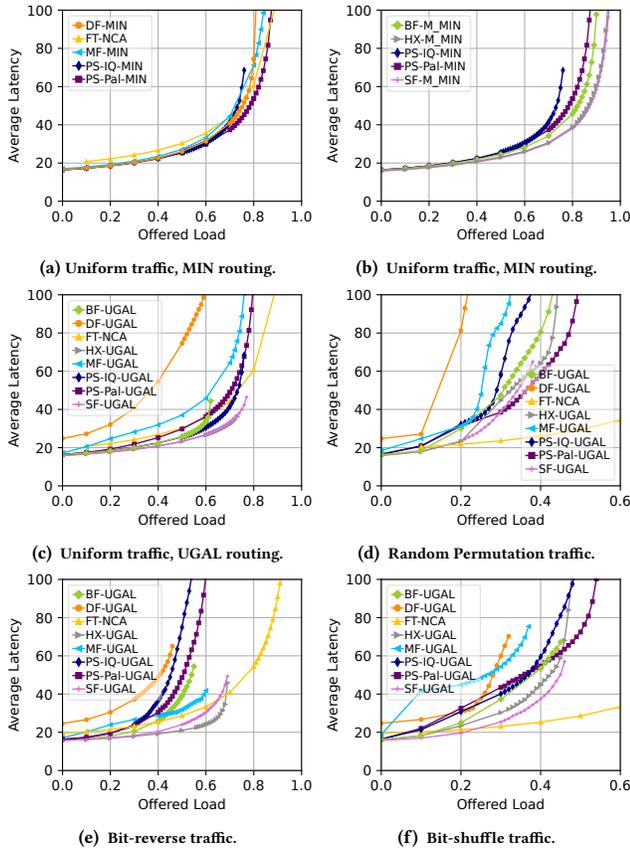


Figure 9: Performance comparison of several topologies. Latency is measured up to the highest injection rate for which simulation is stable. Beyond this, the network is saturated and latency increases with simulation time. In SF and BF, each routing table stores all minpaths to every destination. Unlike other topologies, they exhibit subpar performance when a single minpath is used.

Bit Shuffle performance of MF and DF is poor because they only have one link between each pair of supernodes [23], as opposed to BF and PS-* which have multiple links between the supernodes. This pattern thus highlights the benefits of star-product topologies over DF and MF. Comparatively, MF performs better on Bit Reverse pattern which has more balanced load distribution. For most traffic patterns, HX and SF sustain the highest injection rate. They have a high degree of symmetry, and high link density. However, they lack scaling efficiency (Figure 1) and have 6.7x and 12.8x geometric mean lower scale than PolarStar, respectively.

9.6 Adversarial Traffic

For hierarchical topologies – PS-*, BF, DF and MF, we implement an adversarial pattern where all endpoints in a supernode transmit to endpoints in only one other supernode, resulting in congestion on global (inter-supernode) links. For every source and destination pair, we enforce the longest possible minpath (3-hops in PS-*, BF, DF and MF, and 4-hops in FT), and in PS-* and BF, also the maximum number of global hops (3 in PS-*, 2 in BF). This traffic pattern has been used for worst-case analysis of BF, DF and MF [23, 36, 40]. This may not be the worst-case pattern for PS-*. The true worst-case

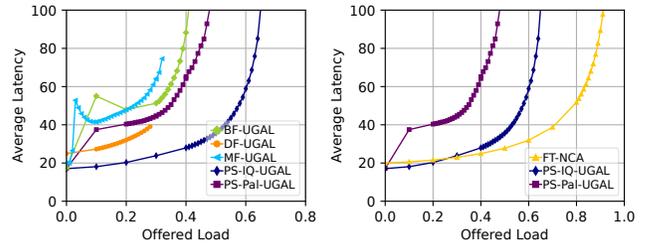


Figure 10: Performance of topologies under adversarial traffic.

may depend on the routing algorithm and the minimum bisection, which is NP-hard to compute for generic graphs.

Figure 10 shows network performance for this traffic pattern. DF and MF saturate at the lowest bandwidth with only a single link between each supernode pair. BF and PS-* perform better because they have multiple links between every supernode pair. PS-IQ performance is better than BF and PS-Pal because of a relatively larger proportion of global links in the particular configuration.

10 EVALUATION: REAL-WORLD MOTIFS

10.1 Simulation Setup

To compare the performance of topologies on real-world motifs, we use the SST simulator [54] and the network configurations listed in Table 3, and construct topologies using SST’s Merlin library [27].

We compare PolarStar (PS-IQ) against the built-in Dragonfly, HyperX and Fat-tree modules in Merlin. We evaluate both MIN and adaptive UGAL routing mechanisms for these topologies. For HyperX, Merlin provides DOAL routing which adaptively routes at most once in each dimension. For simulation, we set router buffer size to 64 KB per port, router and link latencies to 20 ns and link bandwidth to 4 Gbps.

We use two motifs obtained from the Ember Communication Pattern Library [26] of SST:

- *Allreduce* – An important collective in scientific computing, linear solvers and distributed Machine Learning [52, 53, 56].
- *Sweep3D* – This is a wavefront communication pattern moving diagonally on a 2D processor grid. It stresses network latency and is commonly seen in particle simulations and iterative solvers [29].

Message size is 64KB for Allreduce. For both motifs, we measure time for 10 iterations. Process IDs are mapped linearly to endpoints.

10.2 Results

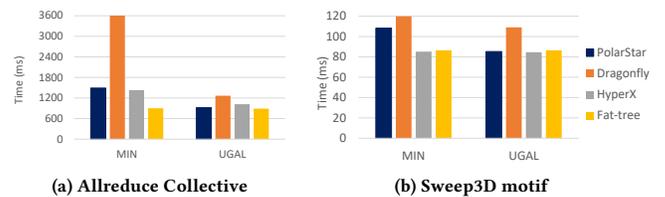


Figure 11: Performance of the Allreduce and Sweep3D motifs.

Allreduce executes fastest on Fat-tree with similar performance on both MIN and adaptive UGAL routing. On the other hand, UGAL performs significantly better than MIN on PolarStar, Dragonfly and HyperX. PolarStar execution time on Adaptive routing is comparable to Fat-tree and better than HyperX. It achieves 2.4× and 1.4× speedup over Dragonfly with MIN and UGAL routing. On Sweep3D, PolarStar is marginally faster than Dragonfly with MIN routing. On UGAL routing, PolarStar’s performance is similar to HyperX and Fat-tree, and 1.28× better than Dragonfly. The improvement over MIN routing is smaller than that seen on Allreduce.

Overall, PolarStar performs better than Dragonfly and comparable to Fat-tree and HyperX. However, it has significantly higher Moore-bound efficiency than HyperX (Figure 1) and can achieve similar scale with smaller switch radix (Table 3).

11 STRUCTURAL ANALYSIS

11.1 Bisection Analysis

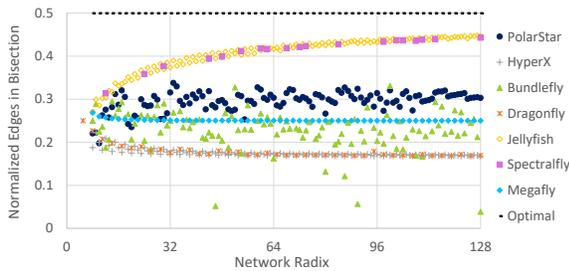


Figure 12: Fraction of links crossing the minimum bisection estimated by METIS [32]. PolarStar, Bundlefly and Spectralfly use their largest feasible (diameter-3) constructions for each radix. Jellyfish has the same radix and scale as PolarStar. Fat-tree and Megafly bisection is normalized by the network links incident with routers that have attached endpoints.

Figure 12 shows the minimum bisection of different topologies for network radix in range [8, 128]. The minimum bisection is estimated using METIS [32] for PolarStar, Spectralfly, Megafly, Bundlefly and Dragonfly. We also compare against Jellyfish which uses a random graph topology [58]. Among the direct networks, Jellyfish has the highest fraction of links in bisection due to the random connectivity between vertices, though its diameter is more than 3. Spectralfly uses Ramanujan graphs that optimize the expansion properties. Hence, it has a large bisection (comparable to Jellyfish), but very few feasible diameter-3 constructions. Among the other diameter-3 topologies, PolarStar has the *highest* proportion of links crossing the bisection, with an average of 29.6% across all radixes. Comparatively, Bundlefly, Dragonfly, HyperX and even the indirect Megafly only have 22.9%, 17.8%, 17.4% and 25.5% links in the bisection cut, respectively. The improved bisection cut can be attributed to the (a) large bisection of the *ER* topology of the structure graph [37], and (b) good radix distribution across supernode and structure graphs due to many choices for supernode radix.

Figure 13 shows the size of bisection cut of PolarStar as a function of radix and supernode topologies. PolarStar with Inductive-Quad and Paley supernodes have an average 29.5% and 26.6% edges in the bisection cut, respectively. The former also offers a more stable bisection across a range of radixes. This is because Inductive-Quad

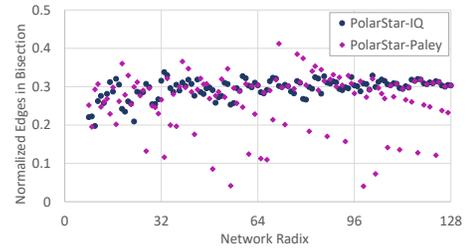


Figure 13: Minimum bisection of PolarStar with Inductive-Quad and Paley supernodes, approximated by METIS [32].

has more feasible radixes and allows better distribution of radix between structure graph and supernode. Comparatively, the limited choice of radixes for Paley graphs may result in a PolarStar with large supernodes and small structure graphs. Such a network will have small bisection because most of the links are concentrated within dense supernode subgraphs.

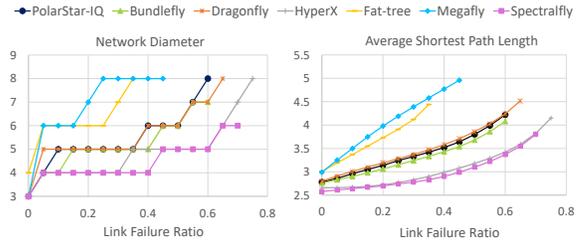


Figure 14: Network Diameter and Average Path Length as a function of random link failures. For Fat-tree and Megafly, we only consider the distance between nodes that have endpoints. PolarStar-Paley behavior is similar to PolarStar-IQ and is not shown for clarity.

11.2 Fault Tolerance

To estimate fault tolerance, we simulate 100 random link failure scenarios until disconnection. We randomly select a simulation with median disconnection ratio, and report the variation in network diameter and average shortest path length in Figure 14.

PolarStar and Bundlefly have similar resilience with a 60% disconnection ratio. Dragonfly has a higher 65% disconnection ratio. However, at low failure ratios, Dragonfly’s diameter and average shortest path length increase rapidly. This is likely because if a global link fails, shortest paths between corresponding groups have to traverse other groups. HyperX and Spectralfly are the most resilient of diameter-3 topologies due to higher connection density, although they suffer from poor scalability (Figure 1). All direct topologies in Figure 14 have a higher disconnection ratio than the indirect topologies Fat-tree and Megafly. Like Dragonfly, Megafly has only one global link between each pair of groups. Therefore, its diameter increases to 6 with just 5% failed links, and its average shortest path length increases faster than the Fat-tree.

REFERENCES

- [1] Marcel Abas. 2017. Large Networks of Diameter Two Based on Cayley Graphs. In *Computer Science On-line Conference*. Springer, 225–233.
- [2] Jung Ho Ahn, Nathan Binkert, Al Davis, Moray McLaren, and Robert S Schreiber. 2009. HyperX: topology, routing, and packaging of efficient large-scale networks. In *Proceedings of the Conference on High Performance Computing Networking, Storage and Analysis*. 1–11.
- [3] Yoshinari Awaji, Kunimasa Saitoh, and Shoichiro Matsuo. 2013. *Optical Fiber Telecommunications VIB: Chapter 13. Transmission Systems Using Multicore Fibers*. Elsevier Inc. Chapters.
- [4] Jun Ho Bahn and Nader Bagherzadeh. 2008. A generic traffic model for on-chip interconnection networks. *Network on Chip Architectures* (2008), 22.
- [5] Eiichi Bannai and Tatsuro Ito. 1973. On finite Moore graphs. *Journal of the Faculty of Science, the University of Tokyo. Sect. 1 A, Mathematics* 20 (1973), 191–208.
- [6] J.C. Bermond, C. Delorme, and G. Farhi. 1982. Large graphs with given degree and diameter III. *Ann. of Discrete Math.* 13 (1982), 23–32.
- [7] Maciej Besta and Torsten Hoefler. 2014. Slim Fly: A cost effective low-diameter network topology. In *SC'14: proceedings of the international conference for high performance computing, networking, storage and analysis*. IEEE, 348–359.
- [8] Maciej Besta and Torsten Hoefler. 2015. Accelerating irregular computations with hardware transactional memory and active messages. In *Proceedings of the 24th International Symposium on High-Performance Parallel and Distributed Computing*. 161–172.
- [9] Mark S Birrittella, Mark Debbage, Ram Huggahalli, James Kunz, Tom Lovett, Todd Rimmer, Keith D Underwood, and Robert C Zak. 2015. Intel® omni-path architecture: Enabling scalable, high performance fabrics. In *2015 IEEE 23rd Annual Symposium on High-Performance Interconnects*. IEEE, 1–9.
- [10] Dhananjay Brahma, Onkar Bhardwaj, and Vipin Chaudhary. 2013. SymSig: A low latency interconnection topology for HPC clusters. In *20th Annual International Conference on High Performance Computing*. IEEE, 462–471.
- [11] W. G. Brown. 1966. On Graphs that do not Contain a Thomsen Graph. *Can. Math. Bull.* 9, 3 (1966), 281–285. <https://doi.org/10.4153/CMB-1966-036-2>
- [12] Cristóbal Camarero, Carmen Martínez, Enrique Vallejo, and Ramón Beivide. 2016. Projective networks: Topologies for large parallel computer systems. *IEEE Transactions on Parallel and Distributed Systems* 28, 7 (2016), 2003–2016.
- [13] Charles Q Choi. 2022. The Beating Heart of the World's First Exascale Supercomputer. <https://spectrum.ieee.org/frontier-exascale-supercomputer>.
- [14] C. Dalfó. 2019. A survey on the missing Moore graph. *Linear Algebra Appl.* (2019).
- [15] R. M. Damerell. 1973. On Moore graphs. *Proc. Camb. Phil. Soc.* 74 (1973), 227–236.
- [16] Hoang-Vu Dang, Roshan Dathathri, Gurbinder Gill, Alex Brooks, Nikoli Dryden, Andrew Lenharth, Loc Hoang, Keshav Pingali, and Marc Snir. 2018. A lightweight communication runtime for distributed graph analytics. In *2018 IEEE International Parallel and Distributed Processing Symposium (IPDPS)*. IEEE, 980–989.
- [17] Stuart Daudlin, Anthony Rizzo, Nathan C Abrams, Sunwoo Lee, Devesh Khilwani, Vaishnavi Murthy, James Robinson, Terence Collier, Alyosha Molnar, and Keren Bergman. 2021. 3D-Integrated Multichip Module Transceiver for Terabit-Scale DWDM Interconnects. In *Optical Fiber Communications, OFC 2021*.
- [18] Aleyah Dawkins, Kelly Isham, Ales Kubicek, Kartik Lakhotia, and Laura Monroe. 2024. Edge-Disjoint Spanning Trees on Star-Product Networks. *arXiv preprint arXiv:2403.12231* (2024).
- [19] Jeffrey Dean and Luiz André Barroso. 2013. The Tail at Scale. *Commun. ACM* 56 (2013), 74–80. <http://cacm.acm.org/magazines/2013/2/160173-the-tail-at-scale/fulltext>
- [20] Jack Dongarra. 2020. *Report on the Fujitsu Fugaku System*. Technical Report ICL-UT-20-06. University of Tennessee, Knoxville.
- [21] Paul Erdős and Alfred Rényi. 1962. On a problem in the theory of graphs. *Publ. Math. Inst. Hungar. Acad. Sci.* 7A (1962), 623–641.
- [22] Paul L. Erdos and Alfréd Rényi. 1963. Asymmetric graphs. *Acta Mathematica Academiae Scientiarum Hungarica* 14 (1963), 295–315.
- [23] Mario Flajslik, Eric Borch, and Mike A Parker. 2018. MegaFly: A topology for exascale systems. In *High Performance Computing (ISC High Performance 2018)*. Springer, 289–310.
- [24] Denis Foley and John Danskin. 2017. Ultra-performance Pascal GPU and NVLink interconnect. *IEEE Micro* 37, 2 (2017), 7–17.
- [25] Bernard Gold and Charles M Rader. 1969. Digital processing of signals. *Digital processing of signals* (1969).
- [26] Simon David Hammond, Karl Scott Hemmert, Michael J Levenhagen, Arun F Rodrigues, and Gwendolyn Renae Voskuilen. 2015. *Ember: Reference Communication Patterns for Exascale*. Technical Report. Sandia National Lab.(SNL-NM), Albuquerque, NM (United States).
- [27] Karl Scott Hemmert. 2018. *Merlin Element Library Deep Dive*. Technical Report. Sandia National Lab.(SNL-NM), Albuquerque, NM (United States).
- [28] A. J. Hoffman and R. R. Singleton. 1960. On Moore Graphs with Diameters 2 and 3. *IBM Journal of Research and Development* 4, 5 (1960), 497–504. <https://doi.org/10.1147/rd.45.0497>
- [29] Adolfo Hoisie, Olaf Lubeck, and Harvey Wasserman. 2007. Performance analysis of wavefront algorithms on very-large scale distributed systems. In *Workshop on wide area networks and high performance computing*. Springer, 171–187.
- [30] Nan Jiang, Daniel U Becker, George Michelogiannakis, James Balfour, Brian Towles, David E Shaw, John Kim, and William J Dally. 2013. A detailed and flexible cycle-accurate network-on-chip simulator. In *2013 IEEE international symposium on performance analysis of systems and software (ISPASS)*. IEEE.
- [31] Christoforos Kachris and Ioannis Tomkos. 2012. A survey on optical interconnects for data centers. *IEEE Communications Surveys & Tutorials* 14, 4 (2012), 1021–1036.
- [32] George Karypis and Vipin Kumar. 2009. MeTis: Unstructured Graph Partitioning and Sparse Matrix Ordering System, Version 4.0. <https://github.com/KarypisLab/METIS>.
- [33] G. Kathareios, C. Minkenber, B. Prisaraci, G. Rodriguez, and Torsten Hoefler. 2015. Cost-Effective Diameter-Two Topologies: Analysis and Evaluation. In *Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis* (Austin, TX, USA). ACM.
- [34] Gordon Keeler. [n. d.]. ERI Programs Panel - Phase II Overview. DARPA ERI Summit 2019.
- [35] John Kim, William J Dally, and Dennis Abts. 2007. Flattened butterfly: a cost-efficient topology for high-radix networks. In *Proceedings of the 34th annual international symposium on Computer architecture*. 126–137.
- [36] John Kim, William J. Dally, Steve Scott, and Dennis Abts. 2008. Technology-Driven, Highly-Scalable Dragonfly Topology. In *Proceedings of the 35th ISCA*. IEEE Computer Society, Washington, DC, USA.
- [37] Kartik Lakhotia, Maciej Besta, Laura Monroe, Kelly Isham, Patrick Iff, Torsten Hoefler, and Fabrizio Petrini. 2022. PolarFly: A Cost-Effective and Flexible Low-Diameter Topology. In *Proceedings of the International Conference on High Performance Computing, Networking, Storage and Analysis*. IEEE, 1–15.
- [38] Kartik Lakhotia, Kelly Isham, Laura Monroe, Maciej Besta, Torsten Hoefler, and Fabrizio Petrini. 2023. In-network Allreduce with Multiple Spanning Trees on PolarFly. In *Proceedings of the 35th ACM Symposium on Parallelism in Algorithms and Architectures*. 165–176.
- [39] Kartik Lakhotia, Fabrizio Petrini, Rajgopal Kannan, and Viktor Prasanna. 2021. Accelerating Allreduce with in-network reduction on Intel PIUMA. *IEEE Micro* 42, 2 (2021), 44–52.
- [40] Fei Lei, Dezun Dong, Xiangke Liao, and José Duato. 2020. Bundlefly: A low-diameter topology for multicore fiber. In *Proceedings of the 34th ACM International Conference on Supercomputing*.
- [41] Fei Lei, Dezun Dong, Xiangke Liao, Xing Su, and Cunlu Li. 2016. Galaxyfly: A Novel Family of Flexible-Radix Low-Diameter Topologies for Large-Scales Interconnection Networks. In *Proceedings of the 2016 International Conference on Supercomputing (Istanbul, Turkey) (ICS '16)*. Association for Computing Machinery, New York, NY, USA, Article 24, 12 pages. <https://doi.org/10.1145/2925426.2926275>
- [42] Charles E Leiserson. 1985. Fat-trees: Universal networks for hardware-efficient supercomputing. *IEEE transactions on Computers* 100, 10 (1985), 892–901.
- [43] Charles E. Leiserson. 1985. Fat-trees: universal networks for hardware-efficient supercomputing. *IEEE Trans. Comput.* 34, 10 (Oct. 1985), 892–901.
- [44] Dongsheng Li, Xicheng Lu, and Jinshu Su. 2004. Graph-theoretic analysis of Kautz topology and DHT schemes. In *IFIP International Conference on Network and Parallel Computing*. Springer, 308–315.
- [45] Lightmatter. [n. d.]. <https://lightmatter.co/>.
- [46] E Loz, H. Pérez-Rosés, and G.Pineda-Villavicencio. 2010. The Degree-Diameter Problem for General Graphs. http://www.combinatoricswiki.org/wiki/The_Degree_Diameter_Problem_for_General_Graphs.
- [47] Robert J. McEliece. 1987. *Finite Fields for Computer Scientists and Engineers*. Springer, Boston, MA.
- [48] Brendan D. McKay, Mirka Miller, and Jozef Širáň. 1998. A note on Large Graphs of Diameter Two and Given Maximum Degree. *Journal of Combinatorial Theory, Series B* (1998).
- [49] Hans Meuer, Erich Strohmaier, Jack Dongarra, Horst Simon, and Martin Meuer. 2023. Top 500: The List. <https://top500.org/lists/top500/>.
- [50] R E Paley. 1933. On Orthogonal Matrices. *Journal of Mathematics and Physics* 12 (1933), 311–320.
- [51] Gregory F Pfister. 2001. An introduction to the infiniband architecture. *High performance mass storage and parallel I/O* 42, 617–632 (2001), 10.
- [52] Karl E Prikopa, Wilfried N Gansterer, and Elias Wimmer. 2016. Parallel iterative refinement linear least squares solvers based on all-reduce operations. *Parallel Comput.* 57 (2016), 167–184.
- [53] Rolf Rabenseifner. 2004. Optimization of collective reduction operations. In *International Conference on Computational Science*, Vol. 3036. 1–9.
- [54] Arun F Rodrigues, K Scott Hemmert, Brian W Barrett, Chad Kersey, Ron Oldfield, Marlo Weston, Rolf Risen, Jeanine Cook, Paul Rosenfeld, Elliot Cooper-Balis, et al. 2011. The structural simulation toolkit. *ACM SIGMETRICS Performance Evaluation Review* 38, 4 (2011), 37–42.
- [55] Hamid Sarbazi-Azad, Mohamed Ould-Khaoua, and Lewis M. Mackenzie. 2001. Communication delay in hypercubes in the presence of bit-reversal traffic. *Parallel*

- Comput.* 27, 13 (2001), 1801–1816.
- [56] Alexander Sergeev and Mike Del Balso. 2018. Horovod: fast and easy distributed deep learning in TensorFlow. *arXiv preprint arXiv:1802.05799* (2018).
- [57] Alexander Shpiner, Zachy Haramaty, Saar Eliad, Vladimir Zdornov, Barak Gafni, and Eitan Zahavi. 2017. Dragonfly+: Low cost topology for scaling datacenters. In *2017 IEEE 3rd International Workshop on High-Performance Interconnection Networks in the Exascale and Big-Data Era (HiPINEB)*. IEEE, 1–8.
- [58] Ankit Singla, Chi-Yao Hong, Lucian Popa, and P. Brighten Godfrey. 2012. Jellyfish: networking data centers randomly. In *Proceedings of the 9th USENIX NSDI* (San Jose, CA). USENIX Association, Berkeley, CA, USA, 17–17.
- [59] Lawrence C Stewart and David Gingold. 2006. A new generation of cluster interconnect. *White Paper, SiCortex Inc* (2006).
- [60] Ola Torudbakken and Ashok V Krishnamoorthy. 2013. A 50Tbps optically-cabled InfiniBand datacenter switch. In *Optical Fiber Communication Conference*. Optica Publishing Group, OTu3H–1.
- [61] Mark Wade, Erik Anderson, Shaha Ardalan, Pavan Bhargava, Sidney Buchbinder, Michael Davenport, John Fini, Haiwei Lu, Chen Li, and Roy Meade. 2020. TeraPHY: a Chiplet Technology for Low-Power, High-Bandwidth In-Package optical I/O. *IEEE Micro* 40, 2 (2020), 63–71.
- [62] Robert Wilber. 1989. Lower bounds for accessing binary search trees with rotations. *SIAM journal on Computing* 18, 1 (1989), 56–67.
- [63] Stephen Young, Sinan Aksoy, Jesun Firoz, Roberto Gioiosa, Tobias Hagge, Mark Kempton, Juan Escobedo, and Mark Raugas. 2022. SpectralFly: Ramanujan Graphs as Flexible and Efficient Interconnection Networks. In *2022 IEEE International Parallel and Distributed Processing Symposium (IPDPS)*. 1040–1050.