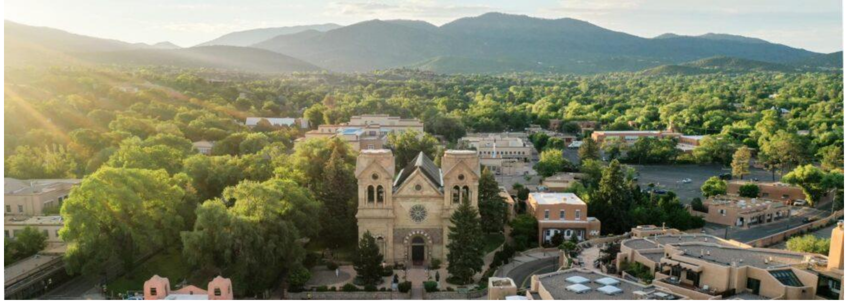
- **Ko Sangre de Christo te maunga.**
  - I come from the Sangre de Cristo subrange of the Rocky Mountains.
- **Nō Santa Fe ahau.**
  - I live in Santa Fe.
- **Ko Rio Grande te awa.**
  - I come from (near) the Rio Grande.
- **Kei Los Alamos National Laboratory ahau e mahi ana.**
  - I work at Los Alamos National Laboratory.
- **Ko Monroe tōku whanau, ko Laura ahau.**
  - My name is Laura Monroe.



- **“Mākoi Pounamu, Tini Putanga kē”**
  - This saying was given to the National Association of Māori Mathematicians, Scientists and Technologists by Prof. Wharehuia Milroy: https://www.taiuru.co.nz/contents-2/
  - This refers to the greenstone tip of a spear which is simple in shape and form but takes time to fashion and is very precious. Although a seemingly simple object, from it comes many different outcomes. It also refers to the scientific bodies of knowledge, the simplicity and complexity and the many outcomes that can be gained from this knowledge.

Math vs. CS
(Good buds that really should hang out more)

Laura Monroe
Ultrascale Systems Research Center
High-Performance Computing Division
Los Alamos National Laboratory

February 20, 2025
Multicore World

Collaborators:
Fabrizio Petrini, Kartik Lakhotia, *Intel Labs*
Kelly Isham , *Colgate University*
Torsten Hoefler, Maciej Besta, Ales Kubicek, *ETH Zurich*
Aleyah Dawkins, Daniel Hwang, *NSF Mathematical Sciences Graduate Internship program*

First some propaganda!

- **Math and CS have a lot to say to each other.**

- **They said it, in great detail, in the middle of the 20th century!**
- **They don't talk so much any more!**
  - Maybe they wave to each other across the room.

- **These days, CS might make use of more mathematical input.**
  - The ongoing demise of Moore's Law.
  - The advent of new computational paradigms (e.g., quantum, AI).
  - The need for more energy efficiency in computing.

What I'm going to talk about

- **An extended example, from networking:**
  - The use of mathematics in network design and in application of networks.

- **In particular, which mathematics approaches really have helped?**
  - ***Classic mathematics*** from graph theory, going back a century.
  - ***A new way of looking at our problem,*** from one of these classic results.
  - The star product: an old graph design, neglected, but very applicable.
    - ***Generalized*** from the extremely non-neglected Cartesian product.
  - And ***we then generalize results*** from the Cartesian case.

- **I'm going to try to give a flavor of this example, NOT in-depth exploration.**
- **Then some discussion about fostering inter-disciplinary communication.**

What I'm not going to talk about

- **Not much about CS network concerns**
  - Bandwidth bisection, routing, etc.
  - These things are really quite good for all networks I discuss.
  - And the mathematical structure helps a lot, for example in routing.

- **There just isn't time.**
- **These discussions can be found in the papers I mention.**

- **I will focus here on this one aspect of a network:**
  - How many nodes can you include, given a radix and a desired diameter?

THE PROBLEM STATEMENT:

Find an optimally large network of a given diameter for routers of a given radix.

Problem: Current networks are not likely to support way past exascale without sacrificing low diameter or other desirable characteristics.

- **We WANT:**
  - Lots of nodes!
  - Low diameter!
  - Exploitable structure!

- **Opportunities:**
  - Near-term routers are large (now up to radix (degree) 512)
  - Need a network that fully uses this connectivity.
  - Photonics offers fast connectivity
  - Need low latency, (low diameter etc) to really exploit this.
- **Constraints:**
  - Mathematical bounds on network size.
  - Difficult to even reaching even those bounds.

The Degree Diameter Problem

- **Again, we want:**
  - As high of a degree (radix) as near-term routers will support.
  - Not many hops = low diameter.
  - Lots of nodes!

- **In other words, this is the degree/diameter problem from graph theory …**
  - Classic graph theory problem.
  - What is the order of the largest graph with a given degree and diameter?
  - The relationship to this problem is good news because a lot of mathematicians have worked on this classical problem and there is a lot of mathematical machinery behind it.
  - Still an open mathematical problem
    - That's good or bad news depending on how you look at it …

Moore's Bound and the Degree-Diameter Problem

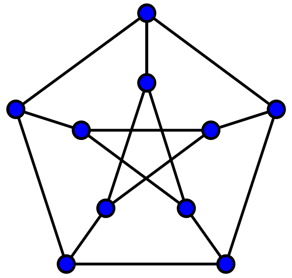
- **The Moore Bound is a mathematical upper bound on the number of nodes given a degree (radix) d and diameter D.**

$$\mathrm{MB}(\mathrm{d},\mathrm{D}) = 1 + d\sum_{i=0}^{D-1}(d-1)^i$$

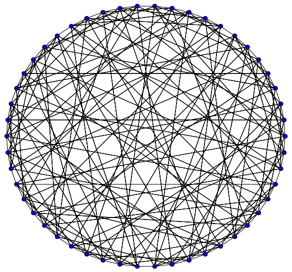
- **The Moore bound is $\mathcal{O}(d^k)$.**
  - For diameter $\mathrm{D} = 1$, the Moore bound is $d + 1$.
  - For diameter $\mathrm{D} = 2$, the Moore bound is $d^2 + 1$.
    - For d = 32, the Moore bound is 1025.
    - For d = 64, the Moore bound is 4097.
    - For d = 128, the Moore bound is 16,385.
  - For diameter $\mathrm{D} = 3$, the Moore bound is $d^3 - d^2 + d + 1$.
    - For d = 32, the Moore bound is 31,777.
    - For d = 64, the Moore bound is 258,113.
    - For d = 128, the Moore bound is 2,080,897.

Do any graphs actually meet the Moore bound?

- **Yes, but not very many, and none good for networking.**
  - The family of complete graphs K_n
  - Degree 2: the odd cycles C_{2n_1}
  - Degree 3: the Petersen graph
  - Degree 7: the Hoffman-Singleton graph
  - Degree 57: May or may not exist.
- **We need to look for asymptotic Moore graphs.**



Petersen graph: degree 3, 10 routers



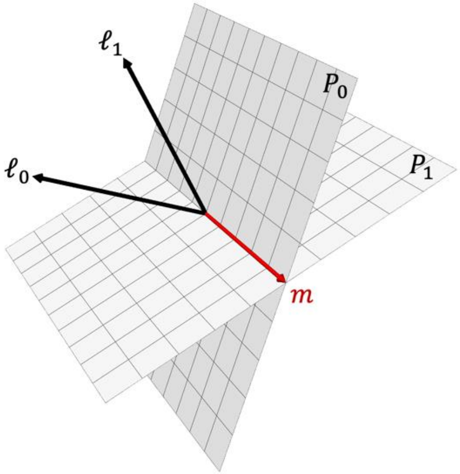
Hoffman-Singleton graph: degree 7, 50 routers

PolarFly (2022):

From the Erdős-Rényi polarity graph (1962)

Basic idea (geometry)

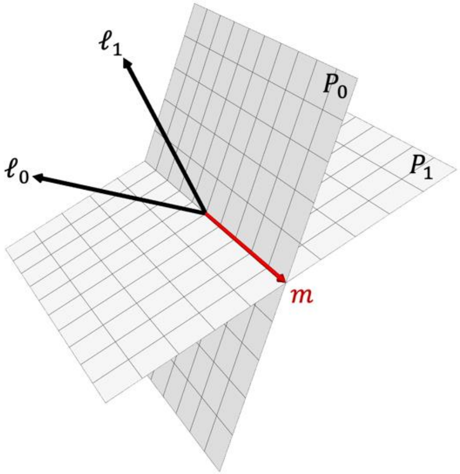
- **If $l_0 \neq l_1$ are any two vectors, there is a vector m perpendicular to both.**
  - (It's the cross-product.)



Basic idea (geometry)

- If $l_0 \neq l_1$ are any two vectors, there is a vector m perpendicular to both.
  - (It's the cross-product.)

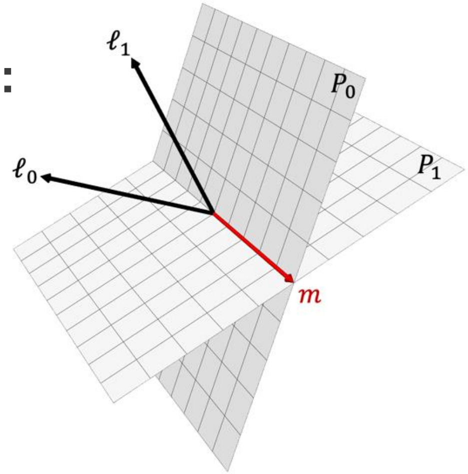
- **What if we constructed a graph with edges expressing dot-product perpendicularity?**
  - (l_0, m) and (m, l_1) are edges in the graph, so you can get from l_0 to l_1 via m.
  - So this graph has diameter $\mathbf{2}$.



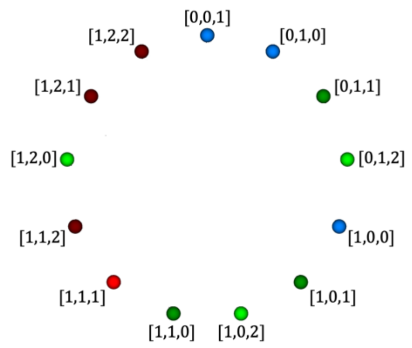
Basic idea (finite geometry … and a little Galois theory)

- If $l_0 \neq l_1$ are any two vectors, there is a vector m perpendicular to both.
  - (It's the cross-product.)
- What if we constructed a graph with edges expressing dot-product perpendicularity?
  - (l_0, m) and (m, l_1) are edges in the graph, so you can get from l_0 to l_1 via m.
  - So this graph has **diameter 2**.

- **Use non-0 3-vectors from $\mathbb{F}_q^3$ whose first non-0 entry is 1:**
  - q is a prime power, and $\mathbb{F}_q$ is the Galois field of order q.
    - For example, if q is a prime, $\mathbb{F}_q$ is just the integers mod q.
  - Fact: each vector is perpendicular to q+1 vectors, so degree is q+1.
  - So the diameter-2 Moore bound is $\boldsymbol{q^2 + 2q + 2}$.
  - Fact: number of nodes/vectors is $\boldsymbol{q^2 + q + 1}$.
- **So this approaches the Moore bound asymptotically !**

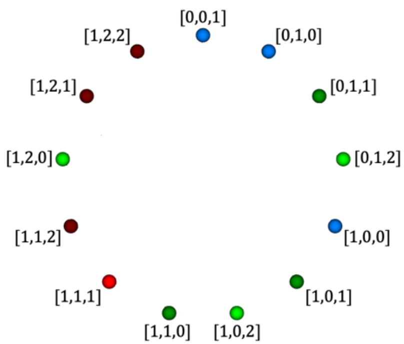


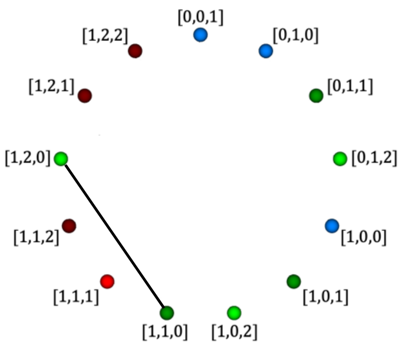
Two layouts of PolarFly



Construction of ER_3

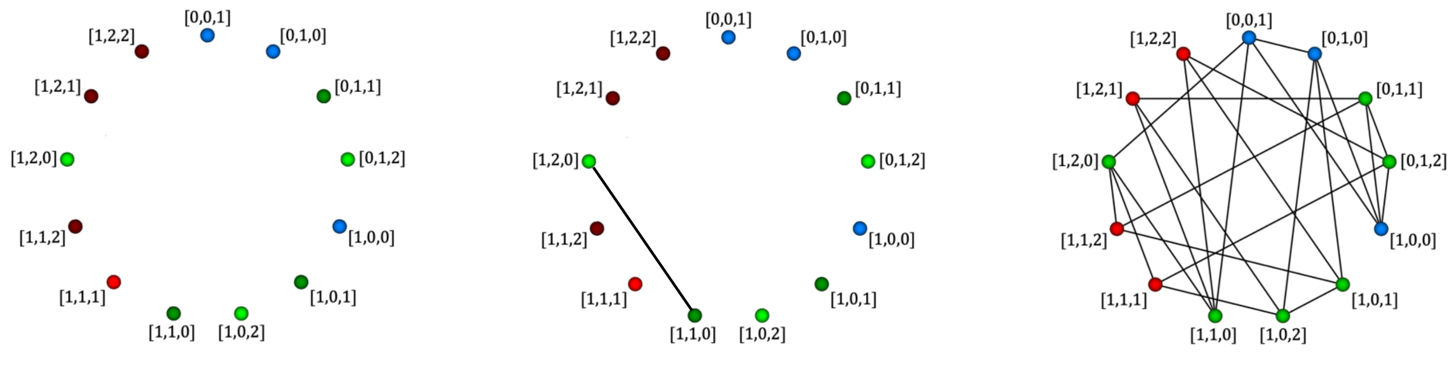
Two layouts of PolarFly





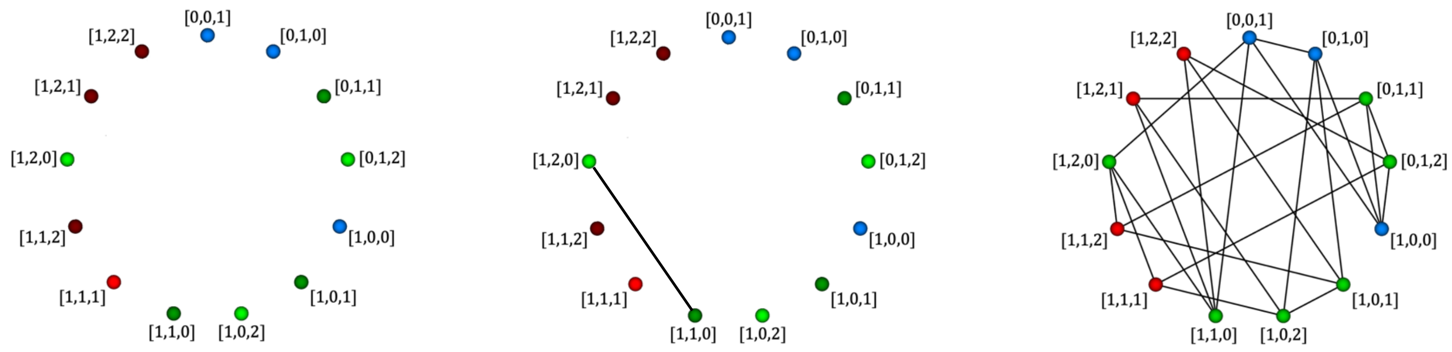
Construction of ER_3

Two layouts of PolarFly

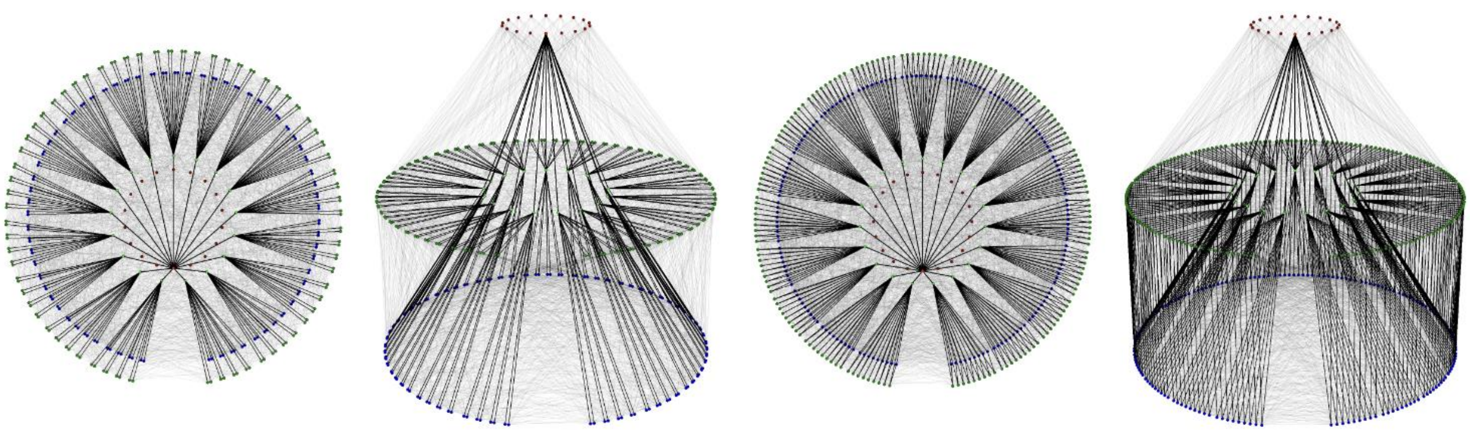


Construction of ER_3

Two layouts of PolarFly



Construction of ER_3



(a) Overhead view of ER_{17}. (b) Side view of ER_{17}. (c) Overhead view of ER_{19}. (d) Side view of ER_{19}.

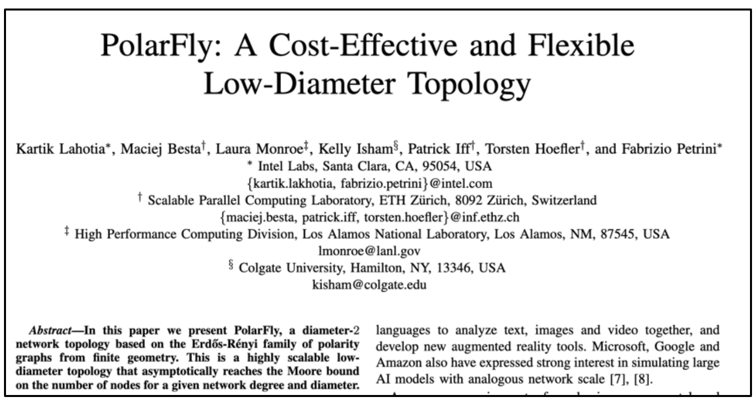
Old math idea, new network.

- **The number of nodes is asymptotically close to the Moore bound ☺**
- **Old math idea (1962).**
  - Independently discovered by Erdős and Rényi, and by Brown.

P. L. Erdos and A. Rényi, "Asymmetric graphs," Acta Mathematica Academiae Scientiarum Hungarica, vol. 14, pp. 295–315, 1963.

W. G. Brown, "On graphs that do not contain a Thomsen graph," Can. Math. Bull., vol. 9, no. 3, p. 281–285, 1966.

- **And we get a nice network out of it.**

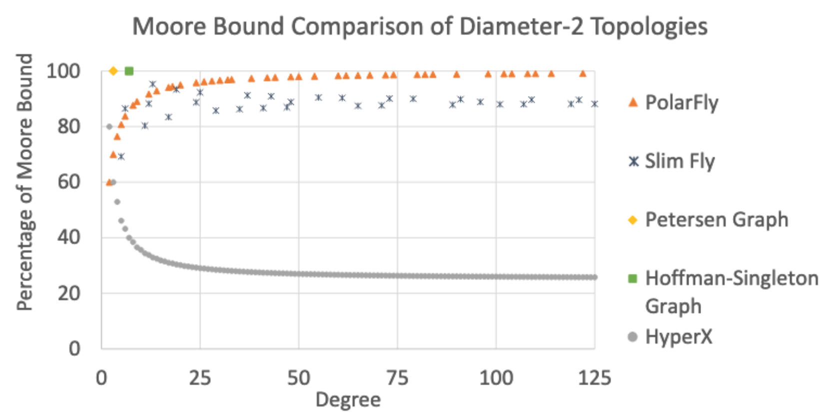


PolarFly: A Cost-Effective and Flexible Low-Diameter Topology

Kartik Lahotia*, Maciej Besta†, Laura Monroe‡, Kelly Isham§, Patrick Iff†, Torsten Hoefler†, and Fabrizio Petrini*
* Intel Labs, Santa Clara, CA, 95054, USA
{kartik.lakhotia, fabrizio.petrini}@intel.com
† Scalable Parallel Computing Laboratory, ETH Zürich, 8092 Zürich, Switzerland
{maciej.besta, patrick.iff, torsten.hoefler}@inf.ethz.ch
‡ High Performance Computing Division, Los Alamos National Laboratory, Los Alamos, NM, 87545, USA
lmonroe@lanl.gov
§ Colgate University, Hamilton, NY, 13346, USA
kisham@colgate.edu

***Abstract*—In this paper we present PolarFly, a diameter-2 network topology based on the Erdős-Rényi family of polarity graphs from finite geometry. This is a highly scalable low-diameter topology that asymptotically reaches the Moore bound on the number of nodes for a given network degree and diameter.**

languages to analyze text, images and video together, and develop new augmented reality tools. Microsoft, Google and Amazon also have expressed strong interest in simulating large AI models with analogous network scale [7], [8].

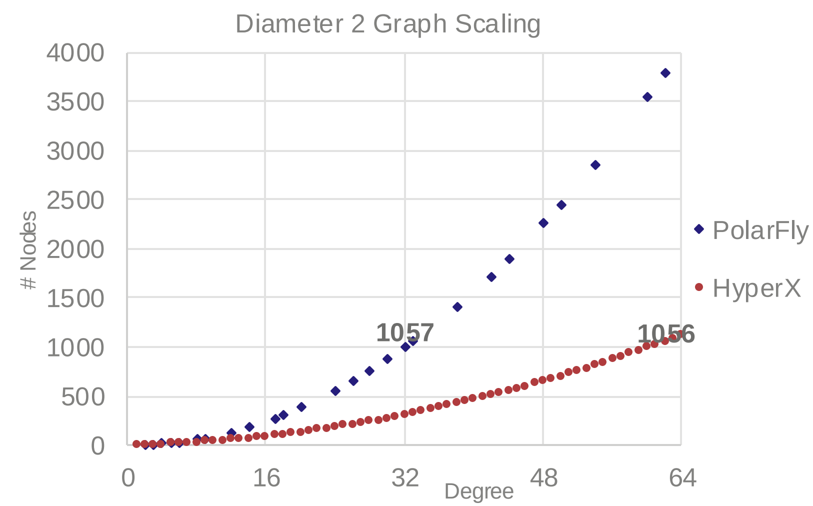


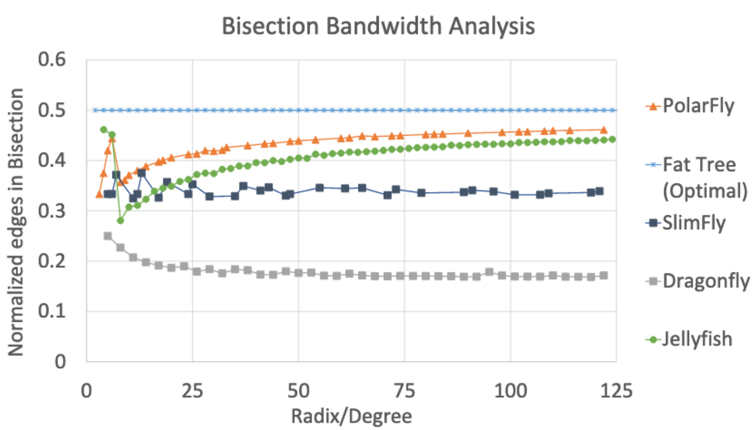
K. Lahotia, M. Besta, L. Monroe, K. Isham, P. Iff, T. Hoefler, and F. Petrini. "PolarFly: A Cost-Effective and Flexible Low-Diameter Topology". The *International Conference for High Performance Computing, Networking, Storage, and Analysis* (SC22). November 2022. https://arxiv.org/abs/2208.01695.

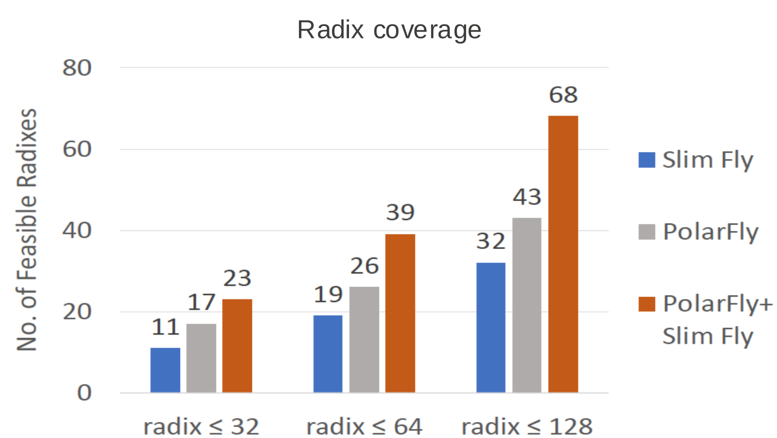
A few metrics on PolarFly

Path diversity

Path length	Conditions	Number of paths
1	v, w adjacent	1
2	v, w adjacent and one of v, w quadric	0
	all other cases	1
3	v, w adjacent	0
	v, w not adjacent, x not quadric	$q-1$
	v, w not adjacent, x quadric	q
4	v, w adjacent and neither of v, w quadric	$(q-1)^2$
	v, w adjacent and one of v, w quadric	q^2-q
	v, w not adjacent and both of v, w quadric	q^2-q
	v, w not adjacent, $v, w \in V_1$, x not quadric	q^2-4
	v, w not adjacent, v quadric, $w \in V_1$	q^2-3
	v, w not adjacent, $v, w \in V_1$, x quadric	q^2-2
	v, w not adjacent, $v \in V_1$, $w \in V_2$	q^2-2
	v, w not adjacent, v quadric, $w \in V_2$	q^2-1
	v, w not adjacent, $v \in V_2$, $w \in V_2$	q^2





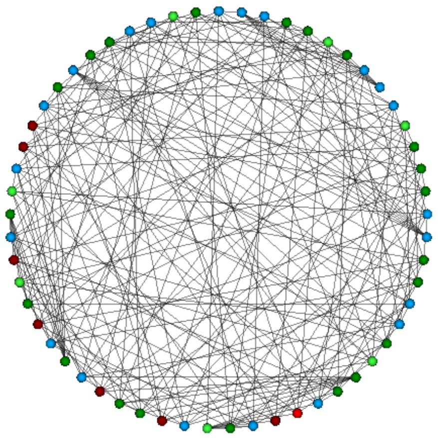


PolarFly (2022):

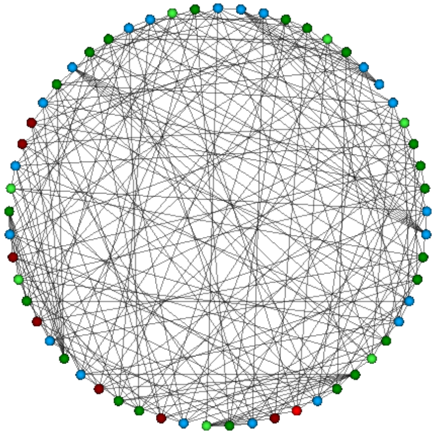
ALSO from the Singer difference set graph (1938)

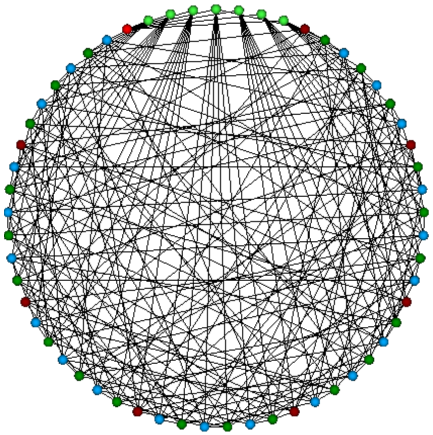
A different and helpful way of looking at it!

Illustrative graph viz: Layouts of ER_7

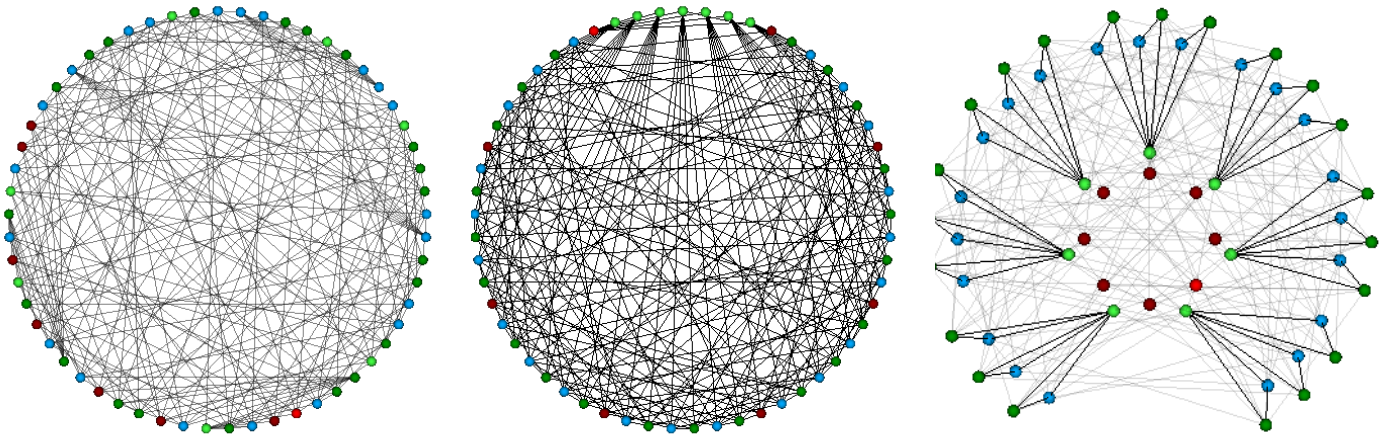


Illustrative graph viz: Layouts of ER_7

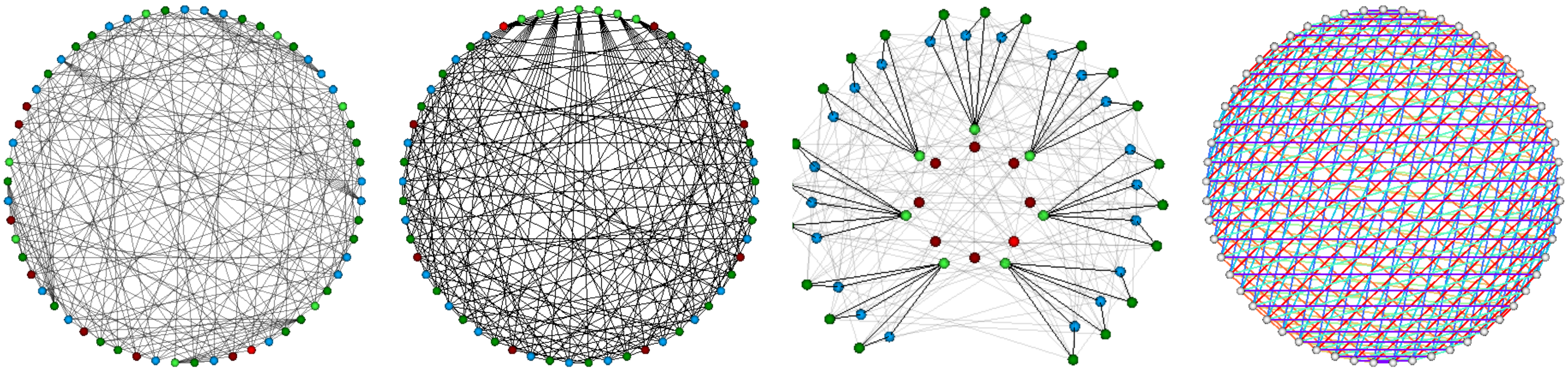




Illustrative graph viz: Layouts of ER_7



Illustrative graph viz: Layouts of ER_7



Singer Difference Sets

- **A subset S of $\mathbb{Z}_N$ with the property that:**
  - For $s_1, s_2 \in S$,
  - Every non-0 element of $\mathbb{Z}_N$ may be expressed as exactly one difference $s_1 - s_2$ mod N.

Singer Difference Sets

- **A subset S of $\mathbb{Z}_N$ with the property that:**
  - For $s_1, s_2 \in S$,
  - Every non-0 element of $\mathbb{Z}_N$ may be expressed as exactly one difference $s_1 - s_2$ mod N.

-	0	1	3	9
0	0	12	10	4
1	1	0	11	5
3	3	2	0	7
9	9	8	6	0

$q = 3$

$N = 9 + 3 + 1 = 13$

Singer Difference Sets

- **A subset S of $\mathbb{Z}_N$ with the property that:**
  - For $s_1, s_2 \in S$,
  - Every non-0 element of $\mathbb{Z}_N$ may be expressed as exactly one difference $s_1 - s_2$ mod N.

-	0	1	3	9
0	0	12	10	4
1	1	0	11	5
3	3	2	0	7
9	9	8	6	0

$q = 3$
$N = 9 + 3 + 1 = 13$

-	0	1	4	14	16
0	0	20	17	7	5
1	1	0	18	8	6
4	4	3	0	11	9
14	14	13	10	0	19
16	16	15	12	2	0

$q = 4$
$N = 16 + 4 + 1 = 21$

Singer Difference Sets

- **A subset S of $\mathbb{Z}_N$ with the property that:**
  - For $s_1, s_2 \in S$,
  - Every non-0 element of $\mathbb{Z}_N$ may be expressed as exactly one difference $s_1 - s_2$ mod N.
- **These always exist for $N = q^2 + q + 1$, with q a prime power.**
  - (Not known if they exist for other N.)
- **How interesting: this reminds us of the Erdős-Rényi graph.**
- **Can we make a graph out of this?**

-	0	1	3	9
0	0	12	10	4
1	1	0	11	5
3	3	2	0	7
9	9	8	6	0

$q = 3$
$N = 9 + 3 + 1 = 13$

-	0	1	4	14	16
0	0	20	17	7	5
1	1	0	18	8	6
4	4	3	0	11	9
14	14	13	10	0	19
16	16	15	12	2	0

$q = 4$
$N = 16 + 4 + 1 = 21$

Yes. A graph may be made from an SDS S:

- Vertices are the elements of $\mathbb{Z}_N$.
- An edge exists between v_1 and v_2 when $v_1 + v_2 \bmod N$ is in S.

Yes. A graph may be made from an SDS S:

- **Vertices are the elements of $\mathbb{Z}_N$.**
- **An edge exists between v_1 and v_2 when $v_1 + v_2$ mod N is in S.**

- **Color the edges:**
  - Assign to each element s of S a color c_s
  - $(v_1\ ,\ v_2)$ is colored c_s if $v_1 + v_2$ mod N is in S.

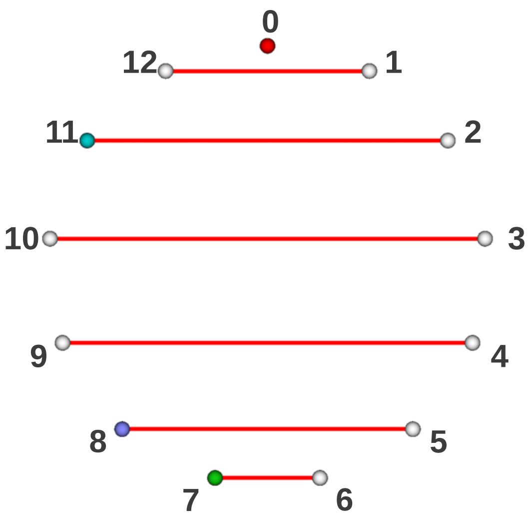
Yes. A graph may be made from an SDS S:

- **Vertices are the elements of $\mathbb{Z}_N$.**
- **An edge exists between v_1 and v_2 when $v_1 + v_2 \bmod N$ is in S.**

- **Color the edges:**
  - Assign to each element s of S a color c_s
  - (v_1, v_2) is colored c_s if $v_1 + v_2 \bmod N$ is in S.

- **Example:** $q = 3,\ N = 13, S = \{0, 1, 3, 9\}$

- **Color the edges $(v_1,\ v_2)$ red if** $v_1 + v_2 = 0\ mod\ 13$.



$S = \{\ 0, 1, 3, 9\ \}$

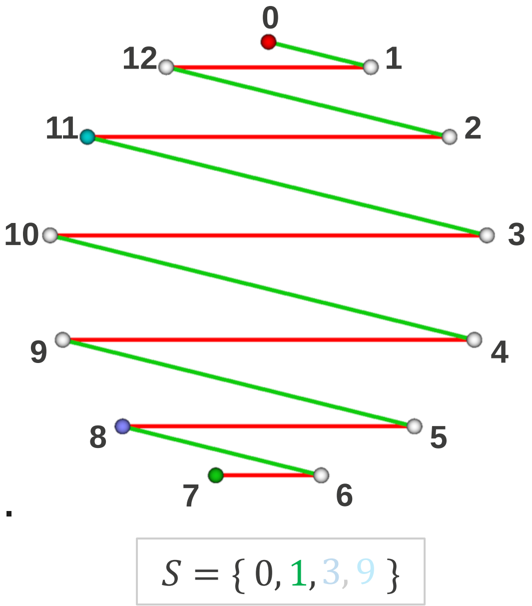
Yes. A graph may be made from an SDS S:

- **Vertices are the elements of $\mathbb{Z}_N$.**
- **An edge exists between v_1 and v_2 when $v_1 + v_2$ mod N is in S.**

- **Color the edges:**
  - Assign to each element s of S a color c_s
  - $(v_1\ , v_2)$ is colored c_s if $v_1 + v_2$ mod N is in S.

- **Example:** $q = 3,\ N = 13, S = \{0, 1, 3, 9\}$

- **Color the edges $(v_1,\ v_2)$ green if** $v_1 + v_2 = 1\ mod\ 13.$



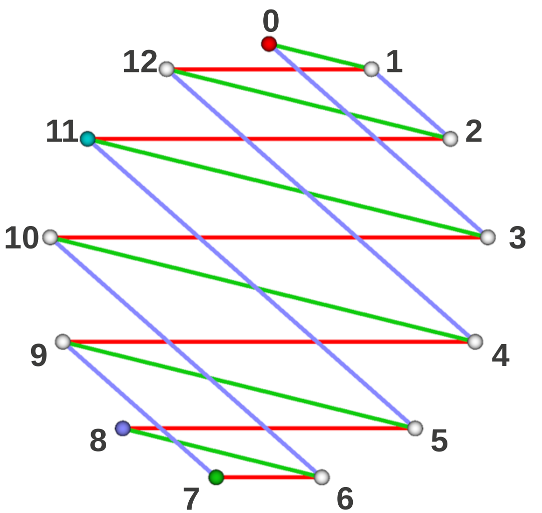
Yes. A graph may be made from an SDS S:

- **Vertices are the elements of $\mathbb{Z}_N$.**
- **An edge exists between v_1 and v_2 when $v_1 + v_2 \bmod N$ is in S.**

- **Color the edges:**
  - Assign to each element s of S a color c_s
  - $(v_1\ , v_2)$ is colored c_s if $v_1 + v_2$ mod N is in S.

- **Example:** $q = 3,\ N = 13, S = \{0, 1, 3, 9\}$

- **Color the edges $(v_1,\ v_2)$ blue if** $v_1 + v_2 = 3\ mod\ 13$.



$S = \{0, 1, 3, 9\ \}$

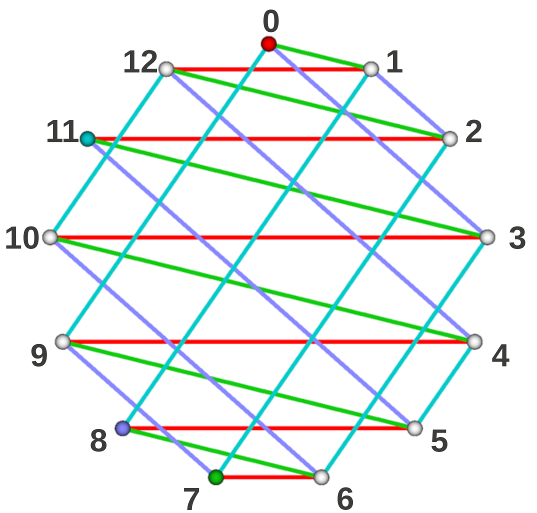
Yes. A graph may be made from an SDS S:

- **Vertices are the elements of $\mathbb{Z}_N$.**
- **An edge exists between v_1 and v_2 when $v_1 + v_2$ mod N is in S.**

- **Color the edges:**
  - Assign to each element s of S a color c_s
  - (v_1 , v_2) is colored c_s if $v_1 + v_2$ mod N is in S.

- **Example:** $q = 3,\ N = 13, S = \{0, 1, 3, 9\}$

- **Color the edges $(v_1,\ v_2)$ cyan if** $v_1 + v_2 = 9\ mod\ 13$.



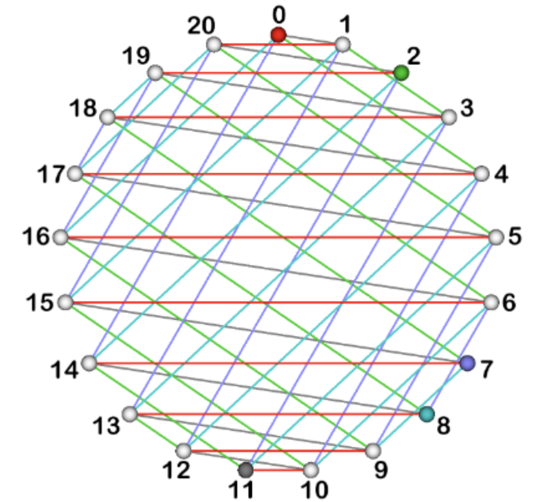
$S = \{0, 1, 3, 9\}$

Yes. A graph may be made from an SDS S:

- **Vertices are the elements of $\mathbb{Z}_N$.**
- **An edge exists between v_1 and v_2 when $v_1 + v_2 \bmod N$ is in S.**

- **Color the edges:**
  - Assign to each element s of S a color c_s
  - $(v_1\ ,\ v_2)$ is colored c_s if $v_1 + v_2$ mod N is in S.

- **Example:** $q = 4,\ \ N = 21, S = \{0, 1, 4, 14, 16\}$



$S = \{\ 0, 1, 4, 14, 16\ \}$

These turn out to be the same as Erdős-Rényi polarity graphs (PolarFly).

- **Singer Difference Sets were introduced by Singer in 1938.**

  James Singer, A Theorem in Finite Projective Geometry and Some Applications to Number Theory. Trans. Amer. Math. Soc. 43, 3, 377–385, 1938.

- **They were used in the Symsig network in 2017.**

- **The associated graphs were shown to be isomorphic to the Erdős-Rényi polarity graphs by Erskine, Fratrič, and Širáň in 2021.**

  Grahame Erskine, Peter Fratrič, and Jozef Širáň, Graphs derived from perfect difference sets. Australas. J Comb. 80 (2021), 48–56, 2021.

- **They are a very interesting and different way of conceptualizing the Erdős-Rényi polarity graphs (PolarFly).**

Edge-disjoint spanning trees on PolarFly

Used in Allreduce, scatter-gather, and other collectives

Edge-disjoint Spanning Trees on PolarFly: Construction

- **Idea: use the colorings of the SDS graph.**
- **Pick two colors c_s and c_t and take the graph consisting of all the edges of those two colors. This graph is:**
  - A spanning tree OR
  - A set of cycles and paths covering all the vertices.

Edge-disjoint Spanning Trees on PolarFly: Construction

- **Idea: use the colorings of the SDS graph.**
- **Pick two colors c_s and c_t and take the graph consisting of all the edges of those two colors. This graph is:**
  - A spanning tree OR
  - A set of cycles and paths covering all the vertices.
- **Clear that if c_a, c_b, c_d and c_e are *different colors*, then the two subgraphs using (c_a, c_b), (c_d, c_e) are *disjoint.***

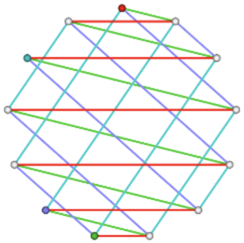
Edge-disjoint Spanning Trees on PolarFly: Our results

- **Idea: use the colorings of the SDS graph.**
- **Pick two colors c_s and c_t and take the graph consisting of all the edges of those two colors. This graph is:**
  - A spanning tree with edges of alternating colors OR
  - A set of cycles and paths with edges of alternating colors, covering all the vertices.
- **Clear that if c_a, c_b, c_d and c_e are *different colors*, then the two subgraphs using (c_a, c_b), (c_d, c_e) are *disjoint.***
- **The graph is a spanning tree iff $c_s - c_t$ is relatively prime to N.**
- **Not only is it a spanning tree, that tree is a** *Hamiltonian path*.

K. Lakhotia, K. Isham, L. Monroe, M. Besta, T. Hoefler, and F. Petrini. 2023. In-network Allreduce with Multiple Spanning Trees on PolarFly. In Proceedings of the 35th ACM Symposium on Parallelism in Algorithms and Architectures (SPAA '23). Association for Computing Machinery, New York, NY, USA, 165–176.

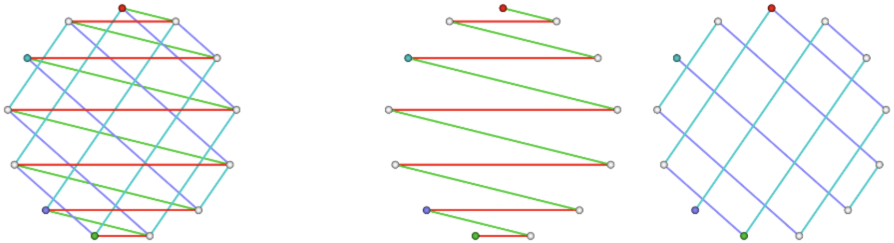
Some examples of edge-disjoint spanning trees

$q = 3$



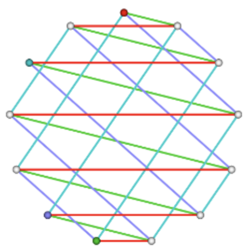
Some examples of edge-disjoint spanning trees

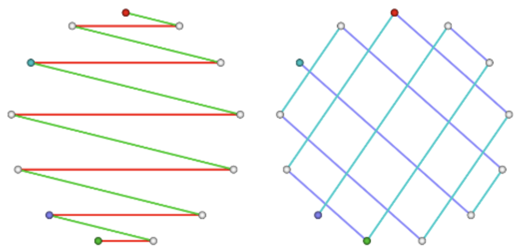
$q = 3$



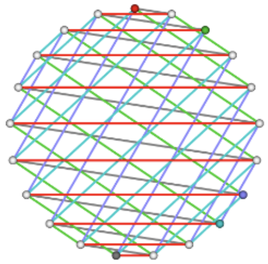
Some examples of edge-disjoint spanning trees

$q = 3$



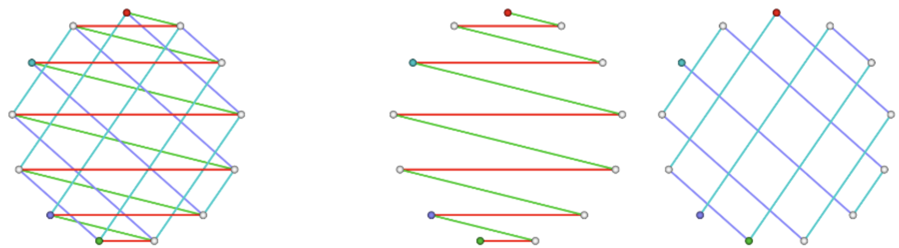


$q = 4$

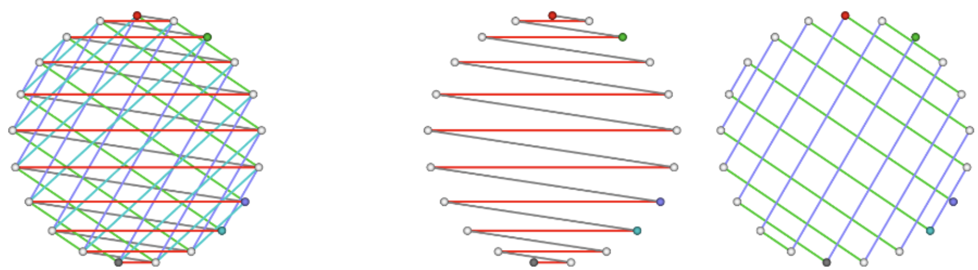


Some examples of edge-disjoint spanning trees

$q = 3$



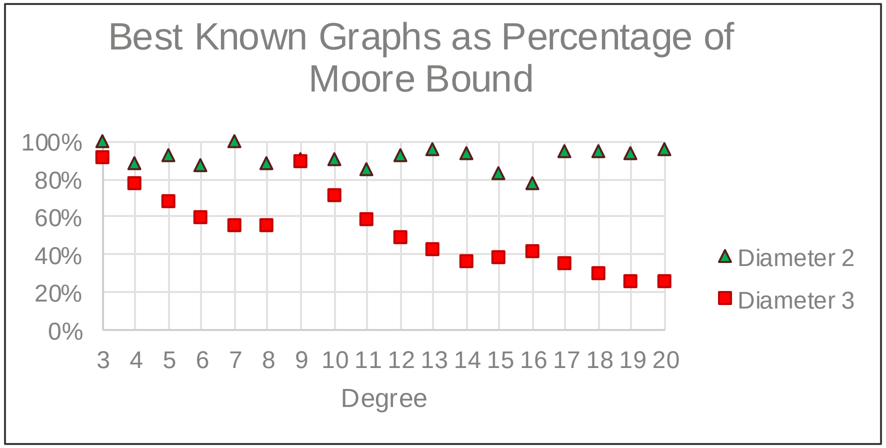
$q = 4$



Cartesian (1637, 1914) and Star Products (1962)

Back to the original problem of large-scale graphs. This time diameter 3.

- **The Moore bound does not seem very tight.**
- **Are there diameter-3 graphs close to the Moore bound?**
  - We (meaning the mathematics and CS community) don't know.
  - In any case, the best known graphs are far short of the bound.

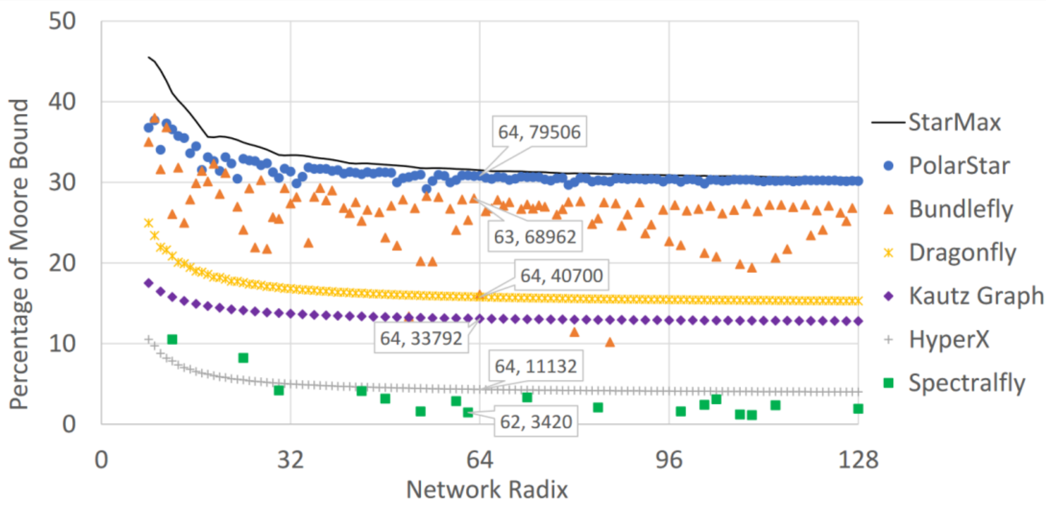


Data from http://combinatoricswiki.org/wiki/The_Degree_Diameter_Problem_for_General_Graphs

Diameter-3 Network Topologies

- **PolarStar: New record size network**
  - ~30% of Moore-bound
  - Star-product extension of PolarFly

Degree	Best known Order in [32]	Moore-bound Efficiency	PolarStar Order	Moore-bound Efficiency
18	1, 620	29.3%	**1, 830**	**33.3%**
19	1, 638	25.1%	**2, 128**	**32.6%**
20	1, 958	25.7%	**2, 394**	**31.4%**



Star products: not new and not fully appreciated

- **Introduced in 1982:**
  - J. Bermond, C. Delorme, and G. Farhi, “Large graphs with given degree and diameter III,” Ann. of Discrete Math., vol. 13, pp. 23–32,1982.
  - Only 32 citations on Semantic Scholar since 1982. (not bad for a math paper)
    - Only 7 directly applied to networking, 2 of them from our group.
  - First use for network design: Bundlefly (2020)
    - 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.

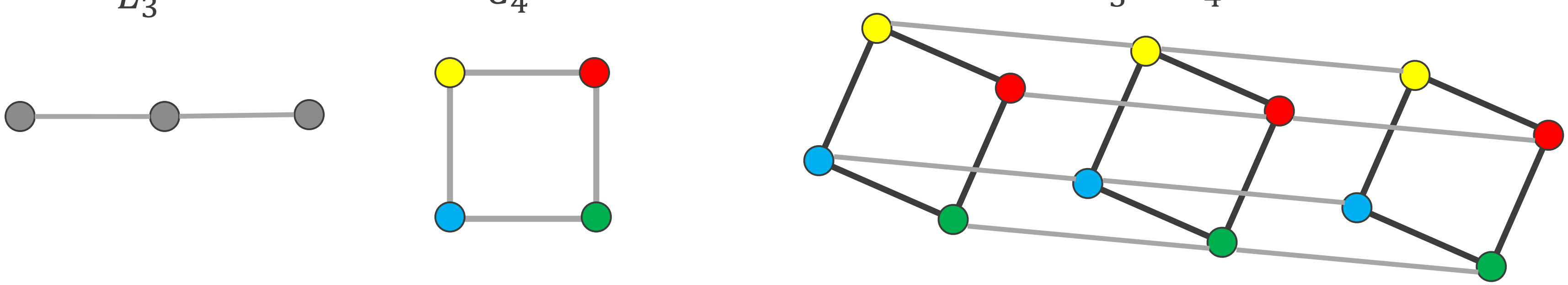
- **Star products generalize Cartesian products**
  - But are much more flexible!
  - A natural for networking
- **One can get diameter only one more than that of one of the factor graphs.**
- **Cartesian graph properties sometimes (maybe often) generalize!**

Quick look at Cartesian products

Example 2.2.2. [Cartesian product] A *Cartesian product* of two graphs G_1 and G_2 is a graph product for which either $a_1 = b_1$ and (a_2, b_2) is an edge in G_2, or $a_2 = b_2$ and (a_1, b_1) is an edge in G_1.

- *Structure graph:* the first factor
- *Supernode:* the second factor

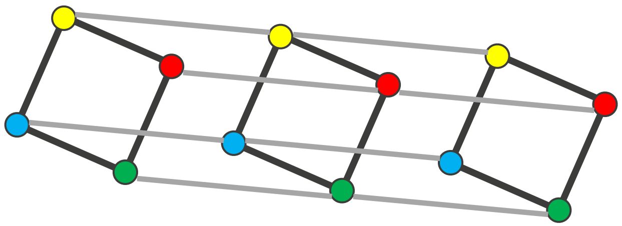
Structure graph L_3 *Supernode* C_4 $L_3 \times C_4$



Join supernodes to each other along the edges of the structure graph.

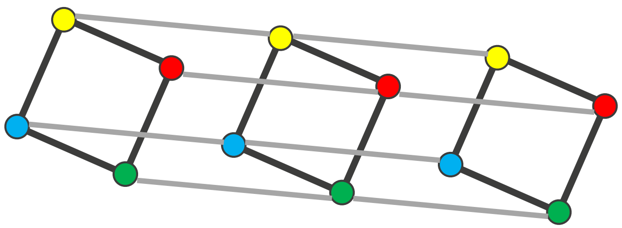
A star product is a generalization of a Cartesian product

- A Cartesian product joins corresponding vertices of the supernodes to each other.

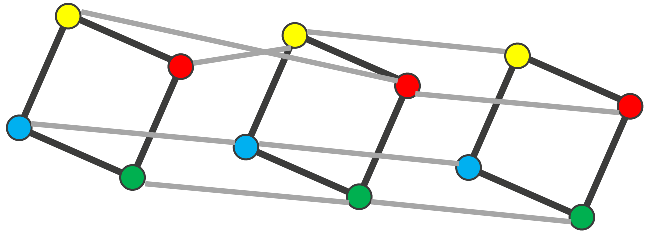


A star product is a generalization of a Cartesian product

- A Cartesian product joins corresponding vertices of supernodes to each other.



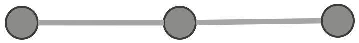
- ***A star product relaxes the "corresponding" constraint.***
- You can join any supernode vertex to neighboring supernode vertices (along edges of structure graph) according to some node permutation f_e.
- Retains the large-scale form of the structure graph.



Star product: informal definition

- Start with a structure graph.

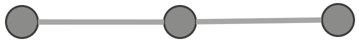
Structure graph
L_3



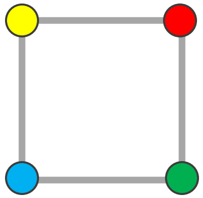
Star product: informal definition

- Start with a structure graph. And a supernode.

Structure graph
L_3



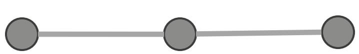
Supernode
C_4



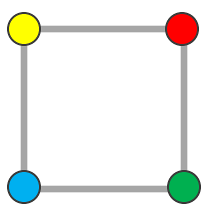
Star product: informal definition

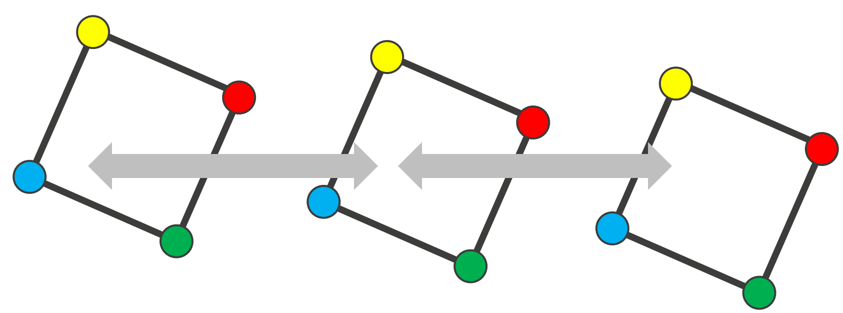
- Start with a structure graph. And a supernode.
- Replace each structure vertex with a copy of the supernode graph.

Structure graph
L_3



Supernode
C_4

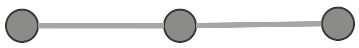




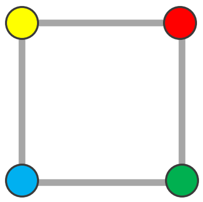
Star product: informal definition

- Start with a structure graph. And a supernode.
- Replace each structure vertex with a copy of the supernode graph.
- Whenever two supernodes are “neighbors”, join their vertices bijectively as you wish and as is convenient.

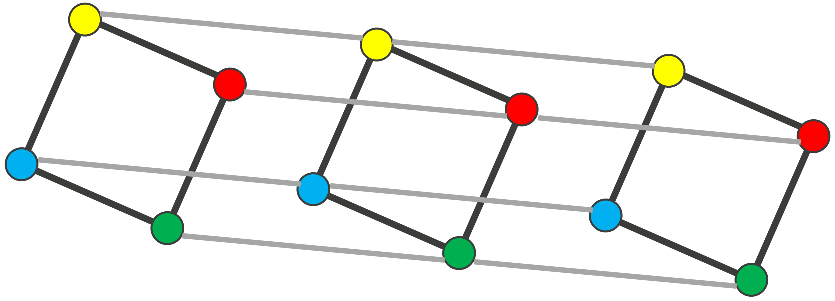
Structure graph
L_3



Supernode
C_4



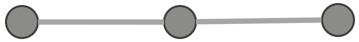
The unique $L_3 \times C_4$



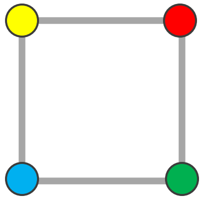
Star product: informal definition

- Start with a structure graph. And a supernode.
- Replace each structure vertex with a copy of the supernode graph.
- Whenever two supernodes are "neighbors", join their vertices bijectively as you wish and as is convenient.

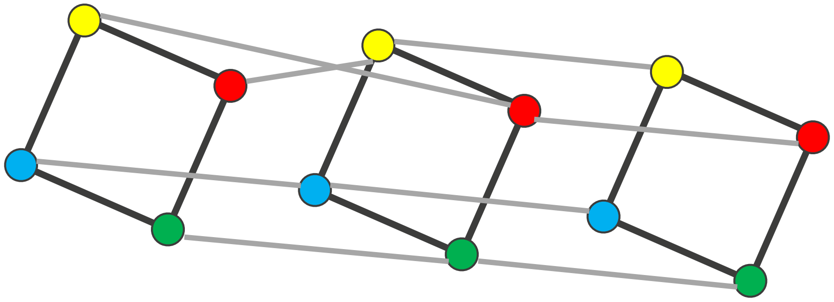
Structure graph
L_3



Supernode
C_4



One of the $L_3 * C_4$



Star product: informal definition

- Start with a structure graph. And a supernode.
- Replace each structure vertex with a copy of the supernode graph.
- Whenever two supernodes are "neighbors", join their vertices bijectively as you wish, and as is convenient.

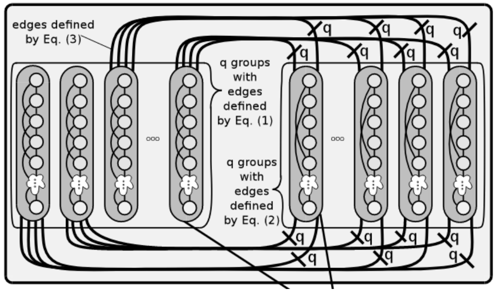
- **Pick the factor graphs right, join the supernodes right:**
- **You get a graph with diameter only 1 more than that of the structure graph!**

J. Bermond, C. Delorme, and G. Farhi, "Large graphs with given degree and diameter III," Ann. of Discrete Math., vol. 13, pp. 23–32,1982.

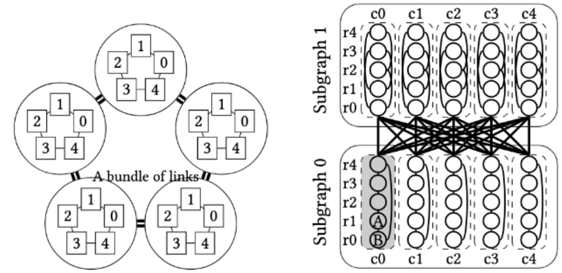
- **Another formulation of "right choices" with best diameter-3 graphs known:**

K. Lakhotia, L. Monroe, K. Isham, M. Besta, N. Blach, T. Hoefler, and F. Petrini, "PolarStar: Expanding the scalability of diameter-3 networks," in Proceedings of the 36th ACM Symposium on Parallelism in Algorithms and Architectures, ser. SPAA '24, 2024, pp. 345–357.

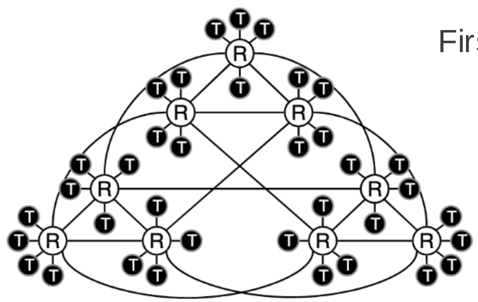
Examples of star products:



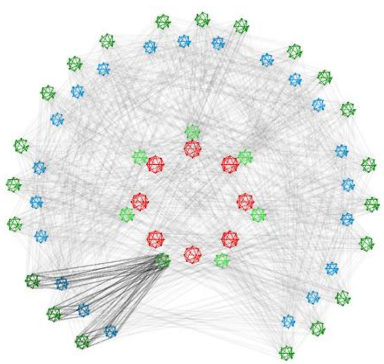
The McKay-Miller-Siran (MMS) graph (1998)
= SlimFly network (2014)
$H_q = K_{q,q} * C(a)$
or $K_{q,q} * QR(a)$
Not previously known to be a star product.



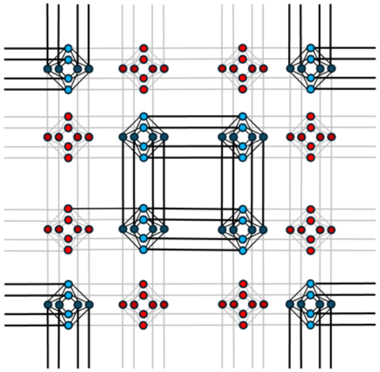
MMS * Paley graph =
BundleFly network (2020)
$H_q * QR(a)$
First explicitly using star product to build networks.



Any Cartesian product is trivially a
star product.
(e.g., HyperX network)



PolarStar network (2024)
$ER_q * QR(a)$
or $ER_q * IQ(d)$
Largest scale family of networks

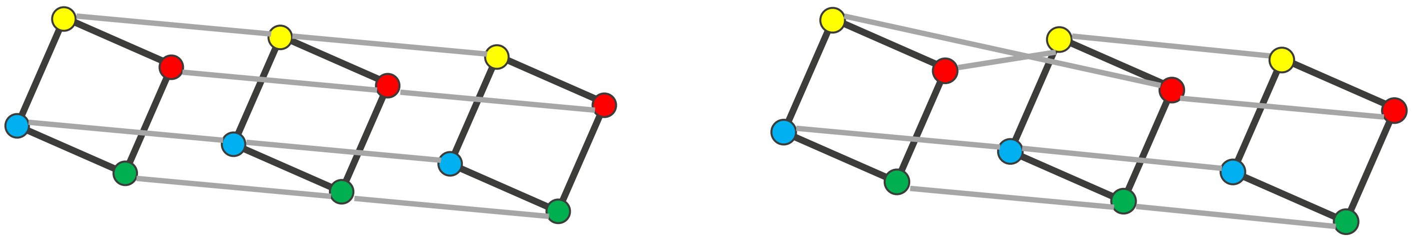


Chimera graph from D-WAVE (2016)
$C_{2(2m+1)} = P_2 * S$
$$S = \left(\bigcup_{2m^2} C_2 \bigcup_{4m} \widehat{K_{4,4}} \bigcup_{2} K_{4,4}\right)$$
Not previously known to be a star product.

Edge-disjoint spanning trees in a star product

Used in Allreduce, scatter-gather, and other collectives

A star product is a generalization of a Cartesian product



Cartesian product A star product

- There is a LOT known about Cartesian products and their associated networks.
- And not that much about star products.
- Maybe we can generalize some of that knowledge?

Edge-disjoint spanning trees maybe?

- **2003 paper from Ku et. al on the construction of EDSTs on Cartesian products, in terms of the EDSTs of the factor graphs.**
- **Should be easy to generalize, knock it right off!**

IEEE TRANSACTIONS ON PARALLEL AND DISTRIBUTED SYSTEMS, VOL. 14, NO. 3, MARCH 2003 213

Constructing Edge-Disjoint Spanning Trees in Product Networks

Shan-Chyun Ku, Biing-Feng Wang, *Member*, *IEEE Computer Society*, and Ting-Kai Hung

Abstract—A Cartesian product network is obtained by applying the cross operation on two graphs. In this paper, we study the problem of constructing the maximum number of edge-disjoint spanning trees (abbreviated to EDSTs) in Cartesian product networks. Let $G = (V_G, E_G)$ be a graph having n_1 EDSTs and $F = (V_F, E_F)$ be a graph having n_2 EDSTs. Two methods are proposed for constructing EDSTs in the Cartesian product of G and F, denoted by $G \times F$. The graph G has $t_1 = |E_G| - n_1(|V_G| - 1)$ more edges than that are necessary for constructing n_1 EDSTs in it, and the graph F has $t_2 = |E_F| - n_2(|V_F| - 1)$ more edges than that are necessary for constructing n_2 EDSTs in it. By assuming that $t_1 \geq n_1$ and $t_2 \geq n_2$, our first construction shows that $n_1 + n_2$ EDSTs can be constructed in $G \times F$. Our second construction does not need any assumption and it constructs $n_1 + n_2 - 1$ EDSTs in $G \times F$. By applying the proposed methods, it is easy to construct the maximum numbers of EDSTs in many important Cartesian product networks, such as hypercubes, tori, generalized hypercubes, mesh connected trees, and hyper Petersen networks.

Index Terms—Cartesian product networks, edge-disjoint trees, spanning trees, embedding, fault-tolerance, interconnection networks.

S.-C. Ku, B.-F. Wang, and T.-K. Hung, "Constructing edge-disjoint spanning trees in product networks," IEEE Transactions on Parallel and Distributed Systems, vol. 14, no. 3, pp. 213–221, 2003

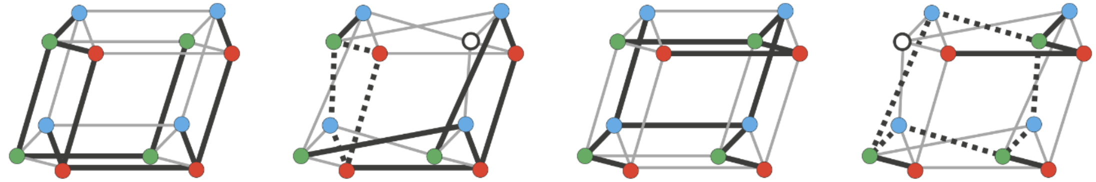
Edge-disjoint spanning trees maybe?

- 2003 paper from Ku et. al on the construction of EDSTs on Cartesian products, in terms of the EDSTs of the factor graphs.
- Should be easy to generalize, knock it right off!

- Why not just apply the bijection to the spanning trees Ku et. al have already constructed?

Edge-disjoint spanning trees maybe?

- **2003 paper from Ku et. al on the construction of EDSTs on Cartesian products, in terms of the EDSTs of the factor graphs.**
- **Should be easy to generalize, knock it right off!**
- **Why not just apply the bijection to the spanning trees Ku et. al have already constructed?**
- **NOPE**



(a) A Cartesian spanning tree (bolded). (b) An f transform of tree edges in 2a. (c) Another Cartesian spanning tree. (d) An f transform of tree edges in 2c.

Eventually, we were able to generalize this result.

Edge-Disjoint Spanning Trees on Star Products

1st Kelly Isham §
Department of Mathematics
Colgate University
Hamilton, NY, USA
kisham@colgate.edu

2nd Laura Monroe §
Ultrascale Research Center
Los Alamos National Laboratory
Los Alamos, NM, USA
lmonroe@lanl.gov

3rd Kartik Lakhotia §
Intel Labs
Intel
Santa Clara, CA, USA
kartik.lakhotia@intel.com

4th Aleyah Dawkins
Department of Mathematics
Carnegie Mellon University
Pittsburgh, PA, USA
adawkins@andrew.cmu.edu

5th Daniel Hwang
Department of Mathematics
Georgia Institute of Technology
Atlanta, GA, USA
dhwang48@gatech.edu

6th Ales Kubicek
Department of Computer Science
ETH Zürich
Zürich, Switzerland
akubicek@student.ethz.ch

***Abstract*—A star-product operation may be used to create large graphs from smaller factor graphs. Network topologies based on star-products demonstrate several advantages including low-diameter, high scalability, modularity and others. Many state-of-the-art diameter-2 and -3 topologies (Slim Fly, Bundlefly, PolarStar etc.) can be represented as star products.**

In this paper, we explore constructions of edge-disjoint spanning trees (EDSTs) in star-product topologies. EDSTs expose multiple parallel disjoint pathways in the network and can be leveraged to accelerate collective communication, enhance fault tolerance and network recovery, and manage congestion.

Our EDSTs have provably maximum or near-maximum cardinality which amplifies their benefits. We further analyze their depths and show that for one of our constructions, all trees have order of the depth of the EDSTs of the factor graphs, and for all other constructions, a large subset of the trees have that depth.

***Index Terms*—Star Product, Spanning Trees, Allreduce**

I. Introduction and Motivation

A. The Star Product in Network Topologies

A network topology can be modeled as a graph connecting nodes (switches or endpoints) using links. One way to build a scalable topology is via a graph product, composing a large graph out of two smaller graphs called *factor graphs*.

We explore here the star product, a family of graphs that has been used to construct some of the largest known low-diameter network topologies. Several new state-of-the-art network topologies are based on the star product [1]–[3].

The star product network generalizes the Cartesian product network (sometimes known simply as the product network), whose networking benefits are well understood [4], [5] and used in popular topologies such as HyperX [6], Torus [7], etc. The star product shares these benefits and structural characteristics, and also improves significantly on the Cartesian product with lower diameter and higher scale.

In this paper, we show a construction of maximum or near maximum sets of edge-disjoint spanning trees (EDSTs) in star product graphs. Such large sets of EDSTs can be used to improve fault-tolerance in large-scale systems, broadcast reliabilty and parallelism in collective operations. Thus, our work adds to the strengths of star-product topologies and makes them more compelling for real-world deployment.

B. Edge-Disjoint Spanning Trees in Networking

A set of spanning trees in a graph G is *edge-disjoint* if no two contain a common edge. Collective operations, especially in-network Allreduces, are typically implemented over spanning trees in the network [8]–[13], using multiple EDSTs to parallelize execution. EDSTs provide independent pathways between all nodes and can be used for reliable communication under component failures or broadcasting system state [14]–[17]. A maximum-sized set of EDSTs maximizes collective bandwidth and fault-tolerance of a system, which improves performance of HPC and Machine Learning workloads [10], [12], [16]–[18]. Thus, a large number of EDSTs is advantageous for any network and generating a maximum-sized set of EDSTs is of great interest (as shown for PolarFly in [12]).

In this paper, we present constructive methods to generate near-maximum or (under certain conditions) maximum-sized sets of EDSTs for star-product graphs using only the EDSTs from smaller and less complex factor graphs.

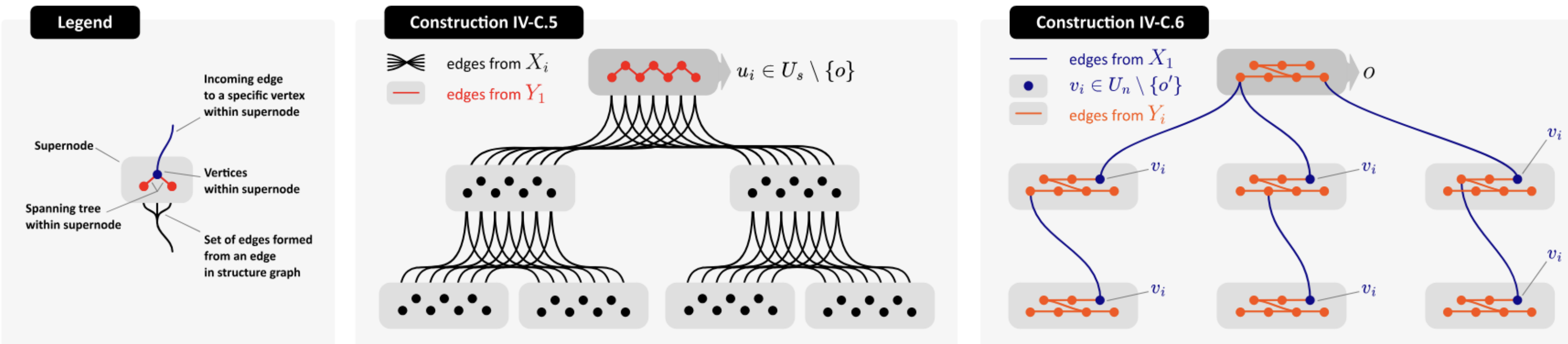
The depth of EDSTs is also critical, as it directly affects the latency and efficiency of communication over the tree. The *depth of a tree* is defined as the maximum path length from any vertex to the root. In this paper, we assume that the *center is the root*. The explicit constructions in this paper enable analysis and optimization of depth and other structural properties of EDSTs. A large fraction of our EDSTs have a depth on the order of the depth of EDSTs in factor graphs, and also present a trade off between cardinality and EDST

This work was supported in part by the Colgate University Picker Interdisciplinary Science Institute major grant, by the U.S. Department of Energy through Los Alamos National Laboratory, operated by Triad National Security, LLC, for the U.S. DOE (Contract # 89233218CNA000001), and by LANL LDRD Project # 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 is assigned LANL identification number LA-UR-24-31132.

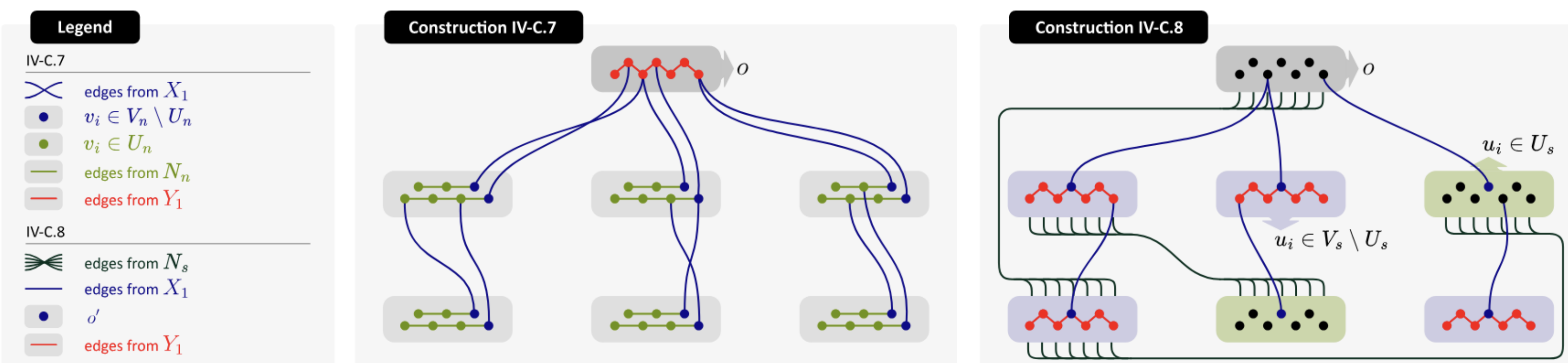
§ The first three authors contributed equally to this work.

K. Isham, L. Monroe, K. Lakhotia, A. Dawkins, D. Hwang and A. Kubíček. (2025). Edge-Disjoint Spanning Trees on Star-Product Networks. To appear at IPDPS 2025.

New EDST constructions.



The first two constructions really did generalize easily, and gave $t_n + t_s - 2$ spanning trees, where t_n (or t_s) is the number of EDSTs in the supernode (or the structure graph.



We could also get up to two more, under the proper conditions. *These were much trickier!*

Summary of results

- **The conditions compare, for each factor graph, t_i (the number of EDSTs in the factor graph) and r_i (the number of remaining edges).**
- **The depth is also calculated in terms of d_i, the depth of the EDSTs in the factor graphs.**

- **The problem reduces to finding EDSTs on smaller simpler factor graphs!**
  - I just discussed a maximum number of EDSTs on ER_q / PolarFly
  - Well known that a maximum number exists on Paley graphs $QR(a)$
  - Well known that a maximum number exists on complete bipartite graphs $K_{q,q}$
  - Believed conjecture: A maximum number exists on $IQ(d)$

Conditions	EDSTs	Max?	EDST Thm.	Max Depth	Min Depth	# Min-Depth EDSTs	Depth Thms.
$r_s = t_s$ AND $r_n = t_n$	$t_s + t_n$	Yes	V-C.3	$(d_{s_1}+1)(2d_{n_1}+2r_s)+d_{s_1}$	$2d_{s_i}+d_{n_1}$	$t_s - 1$	V-C.12, V-C.13
$r_s \geq t_s$ AND $r_n \geq t_n$	$t_s + t_n$	Maybe	V-C.10	$(d_{s_1}+1)(2d_{n_1}+2r_s)+d_{s_1}$	$2d_{s_i}+d_{n_1}$	$t_s - 1$	V-C.12, V-C.13
$r_s \geq t_s$ OR $r_n \geq t_n$	$t_s + t_n - 1$	Maybe	V-C.11	$(d_{s_1}+1)(2d_{n_1}+2r_s)+d_{s_1}$	$2d_{s_i}+d_{n_1}$	$t_s - 1$	V-C.12, V-C.13
Any star product (Universal)	$t_s + t_n - 2$	Maybe	V-B.1	$\max(2d_{s_i}+d_{n_1}, d_{s_1}+2d_{n_i})$	$\min(2d_{s_i}+d_{n_1}, d_{s_1}+2d_{n_i})$	$t_s - 1$ or $t_n - 1$	V-B.4, V-B.5

Summary of results

- **Results:**
  - Generalizations from Ku:
    - You can always make at least $t_n + t_s - 2$ EDSTs.
    - Given conditions that cover most cases, you can make one or two more EDSTs.
  - New results beyond the above generalizations:
    - The upper bound on EDSTs is no more than 2 more than our best construction.
    - In other words, the constructions are linear in terms of EDSTs of the factor graphs
    - AND are also just about the largest number possible.
    - The depths are good too, and if you are careful, you can generate depth linear in terms of the depths of the factor graph EDSTs,

Conditions	EDSTs	Max?	EDST Thm.	Max Depth	Min Depth	# Min-Depth EDSTs	Depth Thms.
$r_s = t_s$ AND $r_n = t_n$	$t_s + t_n$	Yes	V-C.3	$(d_{s_1} + 1)(2d_{n_1} + 2r_s) + d_{s_1}$	$2d_{s_i} + d_{n_1}$	$t_s - 1$	V-C.12, V-C.13
$r_s \geq t_s$ AND $r_n \geq t_n$	$t_s + t_n$	Maybe	V-C.10	$(d_{s_1} + 1)(2d_{n_1} + 2r_s) + d_{s_1}$	$2d_{s_i} + d_{n_1}$	$t_s - 1$	V-C.12, V-C.13
$r_s \geq t_s$ OR $r_n \geq t_n$	$t_s + t_n - 1$	Maybe	V-C.11	$(d_{s_1} + 1)(2d_{n_1} + 2r_s) + d_{s_1}$	$2d_{s_i} + d_{n_1}$	$t_s - 1$	V-C.12, V-C.13
Any star product (Universal)	$t_s + t_n - 2$	Maybe	V-B.1	$\max(2d_{s_i} + d_{n_1}, d_{s_1} + 2d_{n_i})$	$\min(2d_{s_i} + d_{n_1}, d_{s_1} + 2d_{n_i})$	$t_s - 1$ or $t_n - 1$	V-B.4, V-B.5

Remember, these EDST results apply to all star products!

Network	Parameters	$\lvert V_s \rvert$	$\lvert E_s \rvert$	$\lvert V_n \rvert$	$\lvert E_n \rvert$	Upper bound on # EDSTs (Proposition IV-.1)	t_s	r_s	t_n	r_n	Thm.	# EDSTs constructed in this paper	Max?
Slim Fly [1]	$q = 4k+1$	$2q$	q^2	$4k+1$	$k(4k+1)$	$3k$	$2k$	$6k+1$	k	k	V-C.10	$3k$	Yes
$H_q = K_{q,q} * C(q)$	$q = 4k$			$4k$	$4k^2$	$3k$	$2k$	$2k$			V-C.3	$3k$	Yes
[31], [32]	$q = 4k-1$			$4k-1$	$k(4k-1)$	$3k-1$	$2k-1$	$6k-2$			V-C.10	$3k-1$	Yes
Bundlefly [2] $H_q * QR(a)$ [32]	$q = 4\ell+1$, $a = 4k+1$	$2q^2(4k+1)$	$\frac{q^2(3q-1)}{2}$	$4k+1$	$k(4k+1)$	$3\ell+k$	3ℓ	$q^2+3\ell$	k	k	V-C.10	$3\ell+k$	Yes
	$q = 4\ell$, $a = 4k+1$		$\frac{3q^3}{2}$			$3\ell+k$	3ℓ	3ℓ			V-C.3	$3\ell+k$	Yes
	$q = 4\ell-1$, $a = 4k+1$		$\frac{q^2(3q-1)}{2}$			$3\ell+k-1$	$3\ell-1$	$q^2+3\ell-1$			V-C.10	$3\ell+k-1$	Yes
PolarStar [3] $ER_q * QR(a)$ [28], [32]	q even, $a = 4k+1$	q^2+q+1	$\frac{q(q+1)^2}{2}$	$4k+1$	$k(4k+1)$	$\lfloor \frac{q}{2} \rfloor + k$	$\frac{q}{2}$	$\frac{q(q+1)}{2}$	k	k	V-C.10	$\lfloor \frac{q}{2} \rfloor + k$	Yes
	q odd, $a = 4k+1$						$\frac{q+1}{2}$	0			V-C.11	$\lfloor \frac{q}{2} \rfloor + k$	Yes
PolarStar [3] $ER_q * IQ(d)$ [3], [28]	q even, $d = 4m > 0$	q^2+q+1	$\frac{q(q+1)^2}{2}$	$2d+2$	$d(d+1)$	$\lfloor \frac{q+d}{2} \rfloor$	$\frac{q}{2}$	$\frac{q(q+1)}{2}$	$\frac{d}{2}$	$\frac{d}{2}$	V-C.10	$\lfloor \frac{q+d}{2} \rfloor$	Yes
	q even, $d = 4m+3$								$\frac{d-1}{2}$	$\frac{3d+1}{2}$	V-C.10	$\lfloor \frac{q+d}{2} \rfloor$	Yes
	q odd, $d = 4m > 0$						$\frac{q+1}{2}$	0	$\frac{d}{2}$	$\frac{d}{2}$	V-C.11	$\lfloor \frac{q+d}{2} \rfloor$	Yes
	q odd, $d = 4m+3$								$\frac{d-1}{2}$	$\frac{3d+1}{2}$	V-C.11	$\lfloor \frac{q+d}{2} \rfloor - 1$	Maybe

Future Work: other generalizations?

- **We know a LOT about Cartesian product networks.**
  - K. Day and A.-E. Al-Ayyoub, “The cross product of interconnection networks,” IEEE Transactions on Parallel and Distributed Systems, vol. 8, no. 2, pp. 109–118, 1997.
  - A. Youssef, “Cartesian product networks,” in International Conference on Parallel Processing, 1991.
  - Bisection bandwidth
    - J. A. Aroca and A. F. Anta, “Bisection (band)width of product networks with application to data centers,” IEEE Transactions on Parallel and Distributed Systems, vol. 25, no. 3, pp. 570–580, 2014.
  - Path diversity and fault diameter
    - K. Day and A.-E. Al-Ayyoub, “Minimal fault diameter for highly resilient product networks,” IEEE Transactions on Parallel and Distributed Systems, vol. 11, no. 9, pp. 926–930, 2000.
  - Routing algorithms
    - A. Youssef, “Cartesian product networks,” in International Conference on Parallel Processing, 1991.
  - Deadlock avoidance
    - R. Kr´aˇloviˇc, B. Rovan, P. Ruˇziˇcka, and D. ˇStefankoviˇc, “Efficient deadlock-free multi-dimensional interval routing in interconnection networks,” in Distributed Computing, S. Kutten, Ed. Berlin, Heidelberg: Springer Berlin Heidelberg, 1998, pp. 273–287.
  - Resource placement
    - N. Imani, H. Sarbazi-Azad, and A. Zomaya, “Resource placement in Cartesian product of networks,” Journal of Parallel and Distributed Computing, vol. 70, no. 5, pp. 481–495, 2010.

- **Can these be generalized to modern star-product networks?**

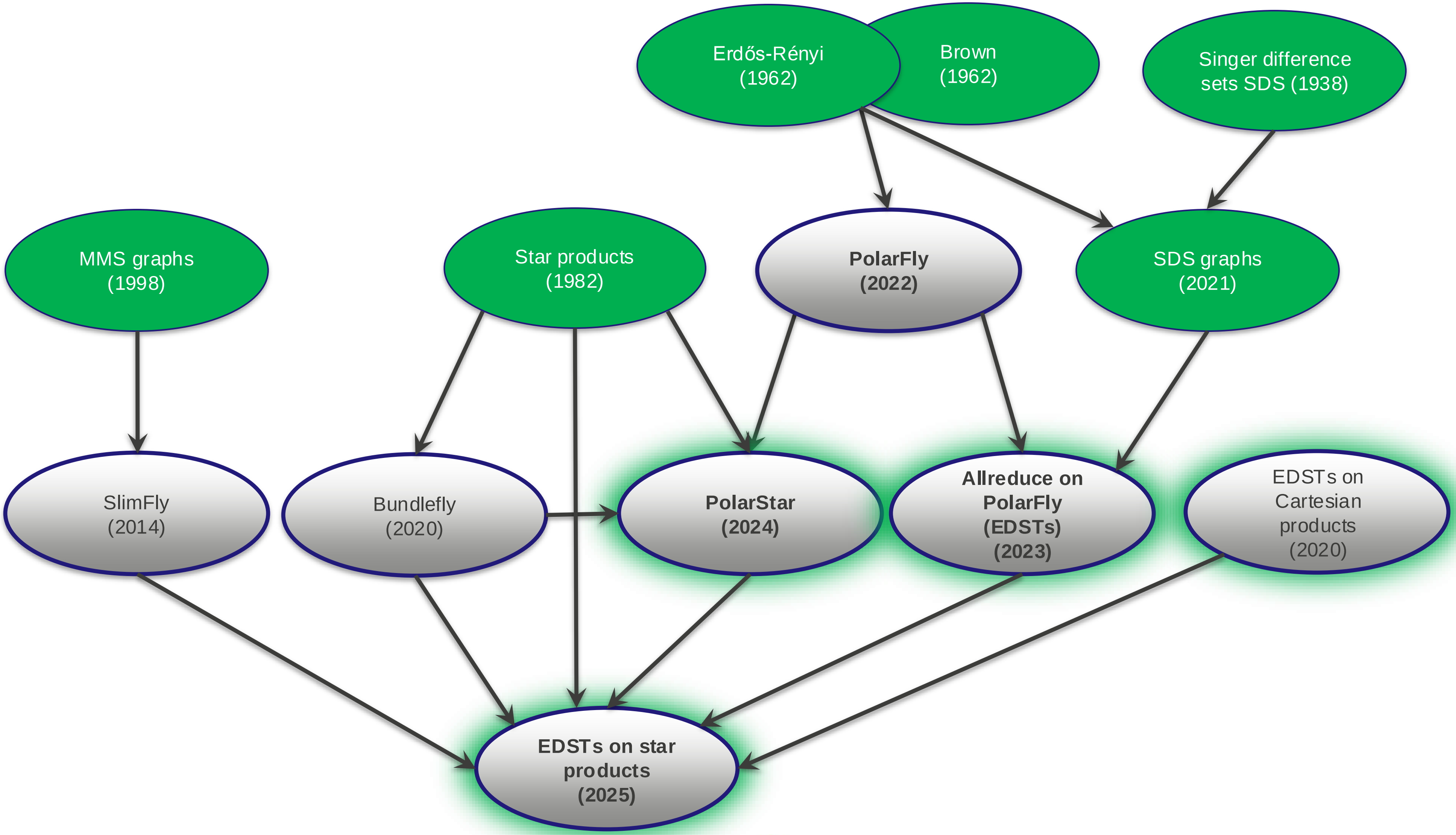
Inter-disciplinary communication

- **Erdős-Rényi polarity graphs**
  - P. L. Erdos and A. Rényi, “Asymmetric graphs,” Acta Mathematica Academiae Scientiarum Hungarica, vol. 14, pp. 295–315, 1963.
  - W. G. Brown, “On graphs that do not contain a Thomsen graph,” Can. Math. Bull., vol. 9, no. 3, p. 281–285, 1966.
  - James Singer, A Theorem in Finite Projective Geometry and Some Applications to Number Theory. Trans. Amer. Math. Soc. 43, 3, 377–385, 1938.
  - Grahame Erskine, Peter Fratrič, and Jozef Širáň, Graphs derived from perfect difference sets. Australas. J Comb. 80 (2021), 48–56, 2021.
- **PolarFly**
  - K. Lahotia, M. Besta, L. Monroe, K. Isham, P. Iff, T. Hoefler, and F. Petrini. “PolarFly: A Cost-Effective and Flexible Low-Diameter Topology”. The International Conference for High Performance Computing, Networking, Storage, and Analysis (SC22). November 2022.
- **Star products**
  - J. Bermond, C. Delorme, and G. Farhi, “Large graphs with given degree and diameter III,” Ann. of Discrete Math., vol. 13, pp. 23–32,1982.
- **Other star products and star-product networks**
  - Brendan D McKay, Mirka Miller, and Jozef Širáň. A note on large graphs of diameter two and given maximum degree. Journal of Combinatorial Theory, Series B, 74(1):110–118, 1998.
  - Paul R. Hafner. Geometric realisation of the graphs of McKay–Miller–Širáň. Journal of Comb. Theory, Series B, 90(2):223–232, 2004.
  - M. Besta and T. Hoefler, “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 (SC14). IEEE, 2014, pp. 348–359.
  - F. Lei, D. Dong, X. Liao, and J. Duato, “Bundlefly: A low-diameter topology for multicore fiber,” in Proceedings of the 34th ACM International Conference on Supercomputing, 2020.
  - K. Lakhotia, L. Monroe, K. Isham, M. Besta, N. Blach, T. Hoefler, and F. Petrini, “PolarStar: Expanding the scalability of diameter-3 networks,” in Proceedings of the 36th ACM Symposium on Parallelism in Algorithms and Architectures, ser. SPAA ’24, 2024, pp. 345–357.
  - Jung Ho Ahn, Nathan Binkert, Al Davis, Moray McLaren, and Robert S. Schreiber. HyperX: topology, routing, and packaging of efficient large-scale networks. Proceedings of the Conference on High Performance Computing Networking, Storage and Analysis, pp. 1–11, 2009.
- **Optimal Spanning Trees and Allreduce on PolarFly and PolarStar**
  - K. Lakhotia, K. Isham, L. Monroe, M. Besta, T. Hoefler, and F. Petrini. 2023. In-network Allreduce with Multiple Spanning Trees on PolarFly. In Proceedings of the 35th ACM Symposium on Parallelism in Algorithms and Architectures (SPAA '23). Association for Computing Machinery, New York, NY, USA, 165–176.
  - K. Isham, L. Monroe, K. Lakhotia, A. Dawkins, D. Hwang and A. Kubíček. (2025). Edge-Disjoint Spanning Trees on Star-Product Networks. To appear in IPDPS 2025.

Research interactions

Math is green

Math-y Computer Science has a green shadow.



Observations on impact of this approach BOTH WAYS

- **New math results.**
  - Largest known diameter-3 graphs.
  - A general solution to edge-disjoint spanning trees on star products.
    - With depth analysis (and they can be built to be shallow)
  - Another new family of graphs beating the above for very limited radixes.

- **Likewise, new Computer Science.**
  - New networks.
  - Large scale.
  - Good routing and bisection bandwidth.
  - Competitive modern networks.

- **IMPORTANT: CS has Good Problems !**

General observations

- **Different vocabulary and mindset.**
  - You do sometimes need an explicit dictionary.
- **You need the assumption of good faith, going both ways.**
- **The disciplines have a different mindset.**
- **Re: publication**
  - It can be hard to place papers in the field.
  - Too math-y for CS conferences, too CS-y for math journals.
  - Author ordering.
- **Very much fun, very productive …**

Outreach between disciplines

- **Joint Mathematics Meetings 2024: Special Session on Network Design and Graph Theory**

Outreach between disciplines

- Joint Mathematics Meetings 2024: Special Session on Network Design and Graph Theory

- Maybe a Dagstuhl seminar?
- Drop me a note if you or your team members would like to attend one.

Thank You

Laura Monroe
lmonroe@lanl.gov