rDAN: Toward Robust Demand-Aware Network Designs

Chen Avin[1] Alexandr Hercules[1] Andreas Loukas[2] Stefan Schmid[3]
[1] Ben-Gurion University, IL [2] EPFL, CH [3] University of Vienna, AT & TU Berlin, DE

Abstract

We currently witness the emergence of interesting new network topologies optimized towards the traffic matrices they serve, such as demand-aware datacenter interconnects (e.g., ProjecToR) and demand-aware peer-to-peer overlay networks (e.g., SplayNets). This paper introduces a formal framework and approach to reason about and design robust demand-aware networks (*DAN*). In particular, we establish a connection between the communication frequency of two nodes and the path length between them in the network, and show that this relationship depends on the *entropy* of the communication matrix. Our main contribution is a novel robust, yet sparse, family of networks, short *rDAN*s, which guarantee an expected path length that is proportional to the entropy of the communication patterns.

1. Introduction

Traditionally, the topologies of computer networks were optimized toward static worst-case criteria, such as maximal diameter, maximal degree, or bisection bandwidth. For example, many modern datacenter interconnects are based on Clos topologies [1] which provide a constant diameter and a high bisection bandwidth. To give another example, peer-to-peer overlay networks are often hypercubic, providing a logarithmic degree and route-length in the worst case.

While such topologies are efficient w.r.t. worst-case traffic patterns, researchers have recently started developing novel network topologies which are optimized towards the traffic matrices which they actually serve, henceforth referred to as *demand-aware networks* (*DAN*) [2, 3, 4]. For example, ProjecToR [2] describes a novel datacenter interconnect based on laser-photodetector edges: these edges can be established according to the served traffic patterns, which have been shown to be far from random but exhibit locality. An example in the context of peer-to-peer networks and distributed data structures are SplayNet overlays [3], whose topology adapts to the peer's interactions.

This paper presents a novel approach to design *robust* demand-aware networks, *rDAN*s, which come with provable performance and robustness guarantees. We study a natural new metric to measure the quality of a demand-optimized network topology, namely whether the provided path lengths are proportional to the *entropy* in the traffic matrix: frequently communicating nodes should be located closer to each other. Entropy is a well-known metric in information and coding theory, and indeed, the network designs presented in this paper are based on coding theory. To this end, we propose a novel robust and sparse family of network topologies which guarantee an entropy-proportional expected path length. Our approach combines two key ideas: (i) the continuous-discrete design introduced in the algorithmic community by Naor and Wieder [5], and (ii) the concept of prefix codes from information theory. The former allows us to formally reason about topologies as well as routing schemes in the continuous space, and a simple discretization results in network topologies which preserve the derived guarantees. The latter enables to relate the topology to the inherent structure (captured by the entropy) of the communication distribution.

More formally, we assume that the network needs to serve route requests that are drawn independently from an arbitrary, but known distribution matrix $\boldsymbol{R}$. That is, the probability of a request from source node u_i to destination node u_j is fixed and given by R_{ij}. Given a network $G = (V, E)$ and a routing algorithm A, denote by $Route_{G,\mathsf{A}}(i, j)$ the path length from node u_i to node u_j according to A. Traditionally, the path lengths are optimized uniformly across all possible pairs. However, we seek to optimize the *expected path length*

$$\mathrm{EPL}(\boldsymbol{R}, G, \mathsf{A}) = \sum_{u_i, u_j \in V} R_{ij} \cdot Route_{G,\mathsf{A}}(i, j). \tag{1}$$

Our main technical contribution is a proof that our designed *rDAN*, $G(\boldsymbol{R})$, and the proposed routing scheme, called CBR (Code-Based Routing), guarantee an expected path length that is a function of an entropy measure of $\boldsymbol{R}$. In particular, the expected path length of the presented *rDAN* is at most the entropy of $\boldsymbol{p}$, $H(\boldsymbol{p})$, where $\boldsymbol{p} = \arg\min\{H(\boldsymbol{p}_s), H(\boldsymbol{p}_d)\}$ and $\boldsymbol{p}_s$ and $\boldsymbol{p}_d$ are the source and destination marginal distributions, respectively, defined as $\boldsymbol{p}_s = \boldsymbol{R}\mathbf{1}$ and $\boldsymbol{p}_d^\top = \mathbf{1}^\top \boldsymbol{R}$; here, $\mathbf{1}$ is the all-ones vector and, being a probability matrix, $\mathbf{1}^\top \boldsymbol{R}\mathbf{1} = 1$. Formally:

Theorem 1. *For any request distribution matrix $\boldsymbol{R}$, the expected path length of CBR on $G(\boldsymbol{R})$ satisfies*

$$EPL(\boldsymbol{R}, G(\boldsymbol{R}), \mathsf{CBR}) < H(\boldsymbol{p}) + 2 \tag{2}$$

We note that, for heavy tailed distributions, the expected path length can be as low as $O(1)$.

In addition, we show that the *rDAN* topologies designed using our approach feature desirable properties along other dimensions. In particular, CBR forwards greedily, i.e., only based on the destination address and the neighbors of the current node. Moreover, our topologies are *robust* although they are also *sparse*. Robustness is quantified with respect to the number of edges that need to be cut in order to disconnect the network. For a set $S \subset V$, the *cut* $C(S, \bar{S})$ is the set of edges connecting S to its complement. Our results are expressed as a function of the cumulative probability of nodes in S, $p_S = \sum_{u_i \in S} p_i$ where $p_i \in \boldsymbol{p}$.

Theorem 2. *For any node set $S \subseteq V$ in $G(\boldsymbol{R})$, such that $p_S \leq 1/2$,*

$$\mathbb{E}\big(|C(S, \bar{S})|\big) = \Omega\left(\frac{np_S}{\log(\min\{p_i\}^{-1})}\right). \tag{3}$$

This theorem states that in order to disconnect a set of nodes in $G(\boldsymbol{R})$, one would need to remove in expectation a number of edges that is proportional to n times the activity level of the set (up to logarithmic factor when $\min\{p_i\}$ is n^{-c} for constant c). Thus, the network does not have bottlenecks.

Our networks have two additional important properties: *sparsity* and *fairness*. Fairness guarantees that the expected degree of a node is proportional to its activity level.

Property 1. *The total number of edges in $G(\boldsymbol{R})$, is at most $4n$.*

Property 2. *The expected out- resp. in-degree of node u_i are $np_i/2 + O(1)$ resp. $np_i + O(1)$.*

It is interesting to note that two classical network topologies (and other similar topologies) with constant average path length, namely the complete graph (*too dense*) and the star graph (*not fair*), do not satisfy *both* properties above.

In terms of related work and novelty, we are not aware of any work on robust network topologies providing entropy-proportional path length guarantees. Moreover, to the best of our knowledge, there is no work on continuous-discrete network designs for non-uniform distribution probabilities.

A practical motivation for our work comes from recent advances in more flexible network designs, also leveraging the often non-uniform traffic demands [6], most notably ProjecToR, but also Helios, REACToR, Flyways, Mirror, Firefly, etc., see [2]. The works closest to ours in terms of approach are the Continuous-Discrete approach [5] and the SplayNet approach [3]. We tailor the former to demand-optimized networks, and provide new insights e.g., on how greedy routing can be used to combine both *forward* and *backward* routing, introducing additional flexibilities. SplayNet focuses on binary search trees, generalizing the classic splay tree datastructures to the distributed

setting. However, unlike the topologies presented in this paper, SplayNets do not provide any robustness or path diversity guarantees.

Bibliographic Note. A technical report of this paper with additional details is available at [7].

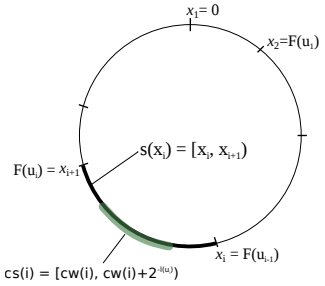
2. Preliminaries

Continuous-Discrete Approach. Our work builds upon the continuous-discrete network design approach introduced by Naor and Wieder [5]. It is based on a discretization of a continuous space into segments, corresponding to nodes. The construction starts with a *continuous graph* G_c defined over a 1-dimensional cyclic space $I = [0, 1)$. For every point $x \in I$, the *left*, *right*, and *backward* edges of x are the points $l(x) = x/2,\ r(x) = (x+1)/2,\ b(x) = \ 2x \bmod 1$, respectively. When x is written in binary form, $l(x)$ effectively inserts a 0 at the left (most significant bit), whereas $r(x)$ shifts a 1 into the left. The backward edge removes the most significant bit. The *discrete network* $G_{\boldsymbol{x}}$ is then a discretization of G_c according to a set of n points $\boldsymbol{x}$ in I, with $x_i < x_{i+1}$ for all i. The points of $\boldsymbol{x}$ divide I into n segments, one for each node: $s_i = [x_i, x_{i+1})\ \forall i < n$ and $s_n = [x_{n-1}, 1) \cup [0, x_1)$. Nodes x_i, x_j are connected by an edge in the discrete graph G if there exists an edge (y, z) in the continuous graph, such that $y \in s_i$ and $z \in s_j$. In addition, we add edges (x_i, x_{i+1}) and (x_{n-1}, x_0) so that $G_{\boldsymbol{x}}$ contains a ring. The authors also noted that their *Distance Halving* construction resembles the well known De Bruijn graphs: if $x_i = \frac{i}{n}$ and $n = 2^r$ then the discrete Distance Halving graph $G_{\boldsymbol{x}}$ without the ring edges is isomorphic to the r-dimensional De Bruijn graph.

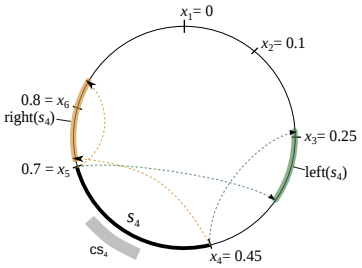
Shannon-Fano-Elias Coding. Shannon-Fano-Elias [8] is a well-known *prefix code* for lossless data compression. We choose this coding method for our network design, due to its simplicity and since its expected code length is optimal (up to a small constant). Consider a discrete random variable of *symbols* X with possible values $\{x_1, \ldots, x_n\}$ and the corresponding symbol probability p_i. The encoding is based on cumulative distribution function (CDF) $F_i = F(x_i) = \sum_{j \le i} p_j$. It encodes symbols using function $\bar{F}_i = \sum_{j<i} p_j + p_i/2 = F_{i-1} + p_i/2$. Denote by $(F)_{01}$ the binary representation of F. The codeword for symbol x_i, denoted as cw_i, consists of the first ℓ_i bits of the fractional part of $\bar{F}_i$, i.e., $cw_i = \lfloor (\bar{F}_i)_{01} \rfloor_{\ell_i}$,where the *code length* ℓ_i is defined as $\ell_i = \lceil \log\ {p_i}^{-1} \rceil + 1$. This construction guarantees (i) that cw_i are prefix-free and (ii) that the expected code length $L_{SFE}(X) = \sum_{i=1}^{n} p_i \cdot l_i = \sum_{i=1}^{n} p_i\ (\lceil \log ({p_i}^{-1}) \rceil + 1)$ is close to the entropy $H(X)$ of random variable X [8]: $H(X) + 1 \le L_{SFE}(X) < H(X) + 2$.

i	p_i	x_i	F_i	$\bar{F}_i$	$(\bar{F}_i)_{01}$	$\ell(i)$	cw_i
1	0.1	0	0.1	0.05	0.000011...	5	00001
2	0.15	0.1	0.25	0.175	0.001011...	4	0010
3	0.2	0.25	0.45	0.35	0.010110...	4	0101
4	0.25	0.45	0.7	0.575	0.100100...	3	100
5	0.1	0.7	0.8	0.75	0.110000...	5	11000
6	0.2	0.8	1.0	0.9	0.111001...	4	1110

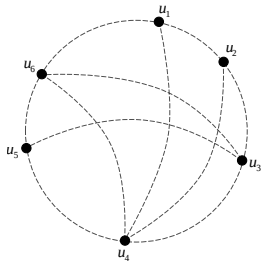
(a)



(b)



(c)



(d)

Figure 1: Illustration of the proposed robust demand-aware network design and its code-based routing.

3. Robust Demand-Aware Network Design

The basic idea behind our robust demand-aware network designs, short *rDAN*, and its Code-Based Routing (CBR), is the following. We start by designing a continuous network G_c in the 1-dimensional cyclic space $I = [0, 1)$. This continuous network is subsequently discretized so as to obtain G. The points $\boldsymbol{x}$ are placed on I according to the CDF of $\boldsymbol{p} = \arg\min_{\boldsymbol{p}_s, \boldsymbol{p}_d}(H(\boldsymbol{p}_s), H(\boldsymbol{p}_d))$. Let U_I be a uniform random variable on I. The i-th point x_i is then given by $x_i = (U_I + F_{i-1} \bmod 1)$ with $i = 1, 2, \ldots, n$. Though adding U_I is not crucial to our construction, the resulting randomness aids us in overcoming an adversary (see later). In the following we will omit the modulo operator, and assume all points $x \in I$ are modulo 1. The rest of the discretization is like in the continuous-discrete approach. In the discrete graph $G_{\boldsymbol{x}}$, each node $u_i \in V$ is associated with the segment $s_i = [x_i, x_{i+1})$, of length p_i. If a point y is in s_i, we say that u_i *covers* y. A pair of vertices u_i and u_j have an edge (u_i, u_j) in $G_{\boldsymbol{x}}$ if there exists an edge (y, z) in the *continuous* graph, such that $y \in s_i$ and $z \in s_j$. The edges (u_i, u_{i+1}) and (u_{n-1}, u_0) are added such that $G_{\boldsymbol{x}}$ contains a ring.

Let us clarify our approach with an example. Consider the activity distribution given in Figure 1 (a) and, for the sake of example, set $U_I = 0$. Carrying out the codeword construction as shown in the table, we obtain the node placement of Figure 1 (b). To discretize the graph, check how the left and right images of each segment intersect other segments. For a segment $s_i = [x_i, x_{i+1})$ its left and right segments are, $l(s_i) = [l(x_i), l(x_{i+1}))$ and $r(s_i) = [r(x_i), r(x_{i+1}))$, respectively. For instance, the left edges of segment s_4 partially cover s_2 and s_3. Therefore, in G the neighbors of node u_4 are u_2 and u_3. In the same way, the right edges of u_4 partially cover s_5 and s_6, which makes the respective nodes neighbors of u_4, see Figure 1 (c). Repeating the same process for all nodes, we obtain the discrete graph $G_{\boldsymbol{x}}$ which is shown in Figure 1 (d).

An important feature of our design *rDAN*s is the relationship between the segment s_i of a node u_i and its codeword cw_i. Let the *ID* of node u_i be $cw_i = \lfloor (\bar{F}_i)_{01} \rfloor_{\ell_i}$ and recall that $\ell_i = \lceil \log\ {p_i}^{-1} \rceil + 1$ is the length of cw_i. Further, define cs_i to be the *code segment* of u_i, $cs_i = [cw_i, cw_i + 2^{-\ell_i})$ and note that cs_i contains all $z \in I$ s.t. cw_i is a prefix of z. It is known from the Shannon-Fano-Elias coding construction that the following relation holds: $cw_i \in cs_i \subseteq s_i$, cf. Figure 1 (c).

Routing. Greedy routing can be performed using two basic methods, *forward* and *backward* routing. Both methods were previously used for fixed length addresses, e.g., in de Bruijn graphs, and thus in our construction require some adjustments due to the variable length of the node IDs.

We start with the ***forward routing*** version of our Code Based Routing (CBR). Recall that cw_i

is the binary code and the ID for u_i and let $cw_i(t)$ denote its suffix of length t. Let $\oplus$ denote the concatenation operator of two strings. For every point $y \in I$ and for every node u_i, we define the function walk$(cw_i(t), y)$ in the following recursive manner: walk$(cw_i(0), y) = y$, walk$(0 \oplus cw_i(t), y) = l(\text{walk}(cw_i(t), y))$, walk$(1 \oplus cw_i(t), y) = r(\text{walk}(cw_i(t), y))$. In other words, walk$(cw_i(t), y)$ is the point reached by a walk of length t that starts at y and proceeds right or left according to the bits of $cw_i(t)$ from its least to most significant bits.

Consider a route from a source u_i to a destination u_j. The starting point of routing is at the source u_i, with $t = 0$ and walk$(cw_j(0), cw_i)$. Upon receiving a message, u_k executes:

Algorithm 1 CBR - Forward Routing in *rDAN* - at node u_k

1: **if** u_k is the destination: **done**.
2: **find** the node u_{next} covering walk$(cw_j(t+1), cw_i) \in s_{\text{next}}$
3: **increase** t and **forward** the message to u_{next}

Similarly, we can define the ***backward routing*** version of CBR (see next) and then claim:

Lemma 1. *For any two nodes, a source u_i and a destination u_j, the* forward (backward) routing *will always reach the destination node. The route length is $\leq \ell_j$ (ℓ_i) hops.*

Proof sketch (for forward routing). First we claim that routing on G_c will reach s_j in ℓ_j hops. The routing starts at cw_i and after ℓ_j steps will reach the point $z = cw_j \oplus cw_i$. Since cw_j is a prefix of z, we have $z \in cs_j \subseteq s_j$, and therefore it is covered by node u_j. To conclude the proof, we note that every hop in the continuous graph between x and y also exists in the discrete graph by definition. □

In the backward routing version of CBR, a message that routes from u_i to u_j starts at $cw_i \oplus cw_j \in cs_i$ and travels the path backwards by removing bits, until reaching $cw_j \in cs_j$. In contrast to the forward routing, the routing path length is at most ℓ_i hops. We can now proceed to proving Theorem 1. Our algorithm CBR uses *forward routing* whenever $H(\boldsymbol{p}_s) \geq H(\boldsymbol{p}_d)$ and *backward routing* when $H(\boldsymbol{p}_s) \leq H(\boldsymbol{p}_d)$.

Proof of Theorem 1. By Lemma 1, for any source u_i and destination u_j, the length of the route is at most the codeword length. Moreover, $\boldsymbol{p}$ is the marginal distribution with minimum entropy and

defines the distribution with which we build the network. The expected path length is therefore:

$$\begin{aligned}\mathrm{EPL}(\boldsymbol{R}, G, \mathsf{CBR}) &= \sum_{u_i,u_j \in V} R_{ij} \cdot Route_{G,\mathsf{CBR}}(i,j) \leq \sum_{u_j,u_i \in V} R_{ij} \cdot \ell_j = \sum_{u_j \in V} \ell_j \sum_{u_i \in V} R_{ij} = \sum_{u_j \in V} p_j \cdot \ell_j \\ &< \sum_{u_j \in V} p_j (\log \frac{1}{p_j} + 2) = H(\boldsymbol{p}) + 2. \end{aligned} \tag{4}$$

□

Finally, let us elaborate on ***routing robustness***. In case of edge failures, our routing algorithms can be continued by sending the message to *any* available neighbor. We add this feature to our algorithms, and when a next hop edge fails at node u_i, we select (independently and uniformly) at random any valid edge to a neighbor node u_j, reset the routing message and send it to u_j, to continue routing as if it is a new route starting from node u_j. To prevent infinite loops we define a maximum routing length restrictions (i.e., TTL).

Network Properties. Let us take a closer look at the basic connectivity properties of the networks designed by our approach. Adapting the proof of Theorem 2.1 in the original Continuous-Discrete paper [5], we can prove that our network is sparse: as Property 1 asserts, the total number of edges in G, without the ring edges, is at most $4n$. Similarly, concerning Property 2, similarly to the original Continuous-Discrete paper [5], (although not formally stated there), the expected node degree is proportional to p_i (which is seen as its activity level).

In addition, the networks designed by our approach are provably robust to edge failures. According to Theorem 2, which is proved next, for any node set $S \subseteq V$ such that $p_S = \sum_{u_i \in S} p_i \leq 1/2$, we have that $\mathbb{E}\big(|C(S,\bar{S})|\big) \geq \Omega(\frac{np_S}{\log p_{\min}^{-1}})$.

Proof of Theorem 2. We use the *expansion* properties of de Bruijn graphs [9]. The *edge expansion* [10] of a graph G is defined as: $h(G) = \min_{0<|S|\leq n/2} C(S,\bar{S})/|S|$. Then for a graph with expansion α and a set S (assume w. l. o. g. that $|S| \leq |\bar{S}|$), the number of edges in the cut is at least $|C(S,\bar{S})| \geq \alpha|S|$. It is known that the expansion of a de Bruijn graph with 2^r nodes is $\Theta(1/r)$ [11]. Our first step will be to bound the image of S in the continuous graph. Let $\mathrm{Im}(S)$ denote the set of points $x \in I$ s.t. x has a neighbor in S in the continuous graph G_c.

Claim 1. $|\mathrm{Im}(S)| = \Theta(\frac{p_s}{\log p_{\min}^{-1}})$

Proof. Let $r = \lfloor \log 3p_{\min}^{-1} \rfloor$ and recall from the preliminaries (Section 2) that if we discretize the continuous graph into uniform size segments of size 2^{-r}, we obtain a de Bruijn graph with 2^r nodes [5]. Denote this graph by G_r and note that the resolution of r guarantees that any S has

$\Theta(p_S/2^r)$ nodes in G_r. Since the expansion of G_r is $\Theta(1/r)$, the edge cut size in G_r is $|C(S,\bar{S})| = \Theta(p_S 2^r/r)$. Now G_r's maximum degree is 4 and the length of each segment is 2^{-r}. □

Regarding $\mathbb{E}\big(|C(S,\bar{S})|\big)$, we need to bound the segments in $\bar{S}$ intersecting with $\mathrm{Im}(S)$. Recall that $p_S \leq 1/2$ and we additionally assume w. l. o. g. that $|S| \leq n/2$ (if this is not the case, we can replace S with $\bar{S}$ to our benefit). $\mathrm{Im}(S)$ may be a union of disjoint segments. Let $s'_1, s'_2, \ldots s'_k$ denote these segments s.t. $\mathrm{Im}(S) = \cup s'_i$ and $|\mathrm{Im}(S)| = \sum |s'_i|$. Assume $\bar{S}$ contains $\ell > n/2$ nodes with corresponding segments $v'_1, v'_2, \ldots v'_\ell$. Let the indicator function $I_{i,j}$ denote whether segment s'_i intersects with segment u'_j. Note that in this case node $u'_j \in \bar{S}$ will have an edge to a node in S. We can now bound $\mathbb{E}\big(|C(S,\bar{S})|\big) = \mathbb{E}\Big(\sum_{i,j} I_{i,j}\Big) = \sum_{i,j} \mathbb{E}(I_{i,j})$. Since v'_j is uniformly distributed in I we have that $I_{i,j} = 1$ w.p. $|s(v'_j)| + |s'_i|$ where $|s(u'_j)|$ is the size of the segment of v'_j, and:

$$\mathbb{E}\big(|C(S,\bar{S})|\big) = \sum_{i,j} |s(v'_j)| + |s'_i| = \sum_{i=1,j=1}^{k,\ell} |s(u'_j)| + \sum_{j=1,i=1}^{\ell,k} |s'_i| \geq \sum_{j=1}^{\ell} |\mathrm{Im}(S)| \geq \frac{n}{2} |\mathrm{Im}(S)| \qquad (5)$$

□

4. Conclusion

This paper introduced a formal metric and approach to design robust and sparse network topologies providing information-theoretic path length guarantees, based on coding. In our future work, we aim to generalize *rDAN*s to other topologies as well as to tailor them to specific use cases (e.g., datacenters).

Acknowledgments. This work was supported by the German-Israeli Foundation for Scientific Research (GIF) Grant I-1245-407.6/2014.

References

[1] M. Al-Fares, A. Loukissas, A. Vahdat, A scalable, commodity data center network architecture, in: ACM SIGCOMM Computer Communication Review, Vol. 38, ACM, 2008, pp. 63–74.

[2] M. Ghobadi, R. Mahajan, A. Phanishayee, N. Devanur, J. Kulkarni, G. Ranade, P.-A. Blanche, H. Rastegarfar, M. Glick, D. Kilper, Projector: Agile reconfigurable data center interconnect, in: Proceedings of the 2016 conference on ACM SIGCOMM 2016 Conference, ACM, 2016, pp. 216–229.

[3] S. Schmid, C. Avin, C. Scheideler, M. Borokhovich, B. Haeupler, Z. Lotker, Splaynet: towards locally self-adjusting networks, IEEE/ACM Transactions on Networking 24 (3) (2016) 1421–1433.

[4] C. Avin, K. Mondal, S. Schmid, Demand-aware network designs of bounded degree, in: Proc. DISC, 2017.

[5] M. Naor, U. Wieder, Novel architectures for p2p applications: the continuous-discrete approach, Vol. 3, ACM, 2007, p. 34.

[6] T. Benson, A. Akella, D. A. Maltz, Network traffic characteristics of data centers in the wild, in: Proceedings of the 10th ACM SIGCOMM conference on Internet measurement, ACM, 2010, pp. 267–280.

[7] C. Avin, A. Hercules, A. Loukas, S. Schmid, Towards communication-aware robust topologies, Arxiv Technical Report `https://arxiv.org/abs/1705.07163`.

[8] T. M. Cover, J. A. Thomas, Elements of information theory, Wiley New York, 2006, Ch. 5, pp. 127–128.

[9] D. N. Bruijn, A combinatorial problem, Proc. Koninklijke Nederlandse Akademie van Wetenschappen. Series A 49 (7) (1946) 758.

[10] S. Hoory, N. Linial, A. Wigderson, Expander graphs and their applications, in: Bulletin AMS, 2006.

[11] F. T. Leighton, Introduction to Parallel Algorithms and Architectures: Array, Trees, Hypercubes, Morgan Kaufmann Publishers Inc., 1992.