The Economics of Decentralized Connectivity*

Paolo Guasoni† Nazem Khan‡ Yufei Zhang§

August 13, 2026

Abstract

What does it cost to maintain short, failure-resilient connectivity in a community without relying on hubs? We construct the Hyperconnected Decentralized Rewiring Architecture (Hydra), an explicit family of networks defined for every hop budget $k \geq 3$ and network size $n \geq k^k$. Hydra is k-step k-connected: every pair of vertices is joined by k internally vertex-disjoint paths of length at most k. Allowing one additional hop raises the guaranteed path multiplicity from k to order $kn^{1/k}$, and Hydra is asymptotically edge-optimal for this stronger hyperconnectivity guarantee. Explicit rewiring rules restore these properties after vertex deletions and insertions.

Hyperconnectivity also produces striking reliability bounds under independent edge failures. For example, with $n = 1{,}000$ and $k = 3$, even if every edge fails independently with probability $1/3$, the probability that a given pair of vertices cannot interact is at most 0.33%. In a maintenance-only benchmark, optimizing the hop budget within the Hydra family gives a near-linear edge count of order $n(\log n/\log\log n)^2$. We also develop a joint model of maintenance, usage, and failure costs that determines the optimal hop budget under power-law activity, and apply the framework to a Hydra routing layer for the Lightning Network. Hydra thereby provides a first step toward an economic theory of decentralized connectivity.

Keywords: connectivity, decentralization, network costs, self-healing networks.

MSC (2020): 90B10, 05C40.

JEL: D85, C61.

*For helpful comments we thank Antoine Jacquier, Eyal Neuman, and seminar participants at the Imperial College and the University of Birmingham.

†Dublin City University, School of Mathematical Sciences, D09 W6Y4 Dublin, Ireland and Università di Bologna, Dipartimento di Statistica, Via Belle Arti 41, 40126 Bologna, Italy. paolo@guasoni.it. Partially supported by Research Ireland (24/FFP-P/12911).

‡The University of Oxford, Mathematical Institute, OX2 6GG Oxford, United Kingdom. nazem.khan@maths.ox.ac.uk.

§Department of Mathematics, Imperial College London, London, UK yufei.zhang@imperial.ac.uk.

1 Introduction

How should a community be organized so that it remains connected when some of its members fail? If failures do occur – whether accidental or malicious – how should it restore its connectivity and fault-tolerance? What is the cost of such robustness?

The difficulty is already visible in the simplest network architectures. A complete graph gives every pair a direct connection and provides abundant alternative routes, but it requires a quadratic number of links. At the other extreme, a star uses only $n-1$ links and has diameter two, but all communication depends on a single hub. A cycle distributes links evenly across its vertices, yet two failures can disconnect it and its diameter grows linearly with the size of the network. Thus neither sparsity, short distances, nor balanced degrees alone provides a satisfactory design.

A robust decentralized architecture must combine several requirements. Every pair of vertices should be connected by multiple routes that use different intermediaries, so that a small number of failures cannot interrupt communication. Those routes should be short because longer chains create more opportunities for delay or failure. The burden of maintaining links should not be concentrated on a small number of hubs, and the total number of links should remain economical. Finally, because participants in operational networks enter and leave, these properties should be recoverable after the network changes rather than guaranteed only by a static initial design.

Payment-channel networks provide a particularly transparent economic example. A direct channel is a costly bilateral commitment involving locked capital, monitoring, and occasional rebalancing, whereas a payment between nonadjacent participants must be forwarded through intermediate channels. Relying on a few highly connected routing nodes can reduce path lengths and the number of channels required, but it also concentrates operational burden, routing power, and vulnerability to targeted disruption. The same design problem arises whenever links are costly standing commitments and exchanges between nonadjacent participants must pass through intermediaries.

Hayek (1945) described competition as "decentralized planning by many separate persons" and viewed the price system as a mechanism for communicating dispersed information. Our question concerns the infrastructure beneath such decentralized coordination. When information or value must pass through a network, can the network itself avoid dependence on a small set of indispensable intermediaries?

Formally, fix a hop budget k and a redundancy target m. A graph is *k-step m-connected* if every pair of distinct vertices is joined by at least m independent – that is, internally vertex-disjoint – paths, each of length at most k. Given n vertices, we study the lexicographic design problem

$$\min\Big\{\big(\Delta(G), \|G\|\big) : |G| = n,\ G \text{ is } k\text{-step } m\text{-connected}\Big\}. \tag{1.1}$$

Here the maximum degree $\Delta(G)$ is minimized first, limiting the largest number of connections assigned to any one vertex and ruling out star-like hubs. Among graphs attaining the optimal maximum degree, the number of edges $\|G\|$ is minimized as a measure of aggregate link cost.

The lexicographic order reflects these design priorities. Connectivity must first be imposed as a feasibility requirement: without it, the empty graph is both costless and perfectly degree-balanced. Conditional on robust short-path connectivity, we place decentralization first because many classical constructions that treat edge count as the primary objective achieve short distances through a small set of hubs; cf. Appendix B.2. Edge count then selects the most economical networks among those attaining the optimal maximum degree.

The primary objective in (1.1) is constrained by a classical degree-diameter trade-off. Every k-step m-connected graph has diameter at most k. If its maximum degree is $\Delta \geq 2$, then the

Moore bound gives

$$n \le 1 + \Delta \sum_{j=0}^{k-1} (\Delta - 1)^j \le 1 + \Delta^k,$$

and hence $\Delta \ge (n-1)^{1/k}$. Thus, even without using the full redundancy requirement, a fixed hop budget forces the largest degree to grow at least on the scale $n^{1/k}$. In particular, no architecture can simultaneously bound every vertex degree independently of n, and guarantee communication within k steps between every pair. The central question is whether this unavoidable degree scale can be attained while also providing multiple independent routes, using close to the minimum possible number of links, and remaining maintainable as vertices enter and leave.

The answer is yes. The starting point is a coordinate grid. Given side lengths $q_1, \dots, q_k \ge 3$, place one vertex at each point of $\{1, \dots, q_1\} \times \cdots \times \{1, \dots, q_k\}$. Two vertices are adjacent if they differ in exactly one coordinate, so every vertex has the same degree $M_H := \sum_{\ell=1}^{k} (q_\ell - 1)$. This is the standard k-dimensional Hamming graph $H(q_1, \dots, q_k)$.

Its coordinate structure generates many short paths. To travel from $x = (x_1, \dots, x_k)$ to $y = (y_1, \dots, y_k)$, one may correct their differing coordinates in different orders, while unused coordinate values provide additional detours. Proposition 5.10 shows that the graph is k-step k-connected. Moreover, Proposition E.2 shows that it is $(k+1)$-step M_H-connected.

Writing $N := \prod_{\ell=1}^{k} q_\ell$ for the number of vertices, the Hamming graph has $NM_H/2$ edges and is edge-optimal for its $(k+1)$-step M_H-connectivity guarantee. When the side lengths are balanced, that is $q_i \approx N^{1/k}$, then $M_H = \Theta(kN^{1/k})$. Thus the balanced Hamming structure already combines short-path resilience, degree-scale hyperconnectivity, decentralization, and edge economy.

Hydra construction. The Hamming benchmark already possesses every static property we seek at balanced product sizes. However, it contains exactly $\prod_{\ell=1}^{k} q_\ell$ vertices, with the perfectly symmetric case given by $N = q^k$, whereas an arbitrary prescribed network size need not admit a balanced product representation. In particular, this class is not closed under single-vertex arrivals and departures.

Hydra removes this restriction. Given an arbitrary n, it begins with a near-balanced ambient Hamming graph on $Q := \prod_{\ell=1}^{k} q_\ell \ge n$ vertices and deletes the $Q - n$ surplus vertices from a designated reference slice. Because deletion destroys some of the coordinate routes, Hydra adds two classes of auxiliary edges: anchor-repair edges transfer the lost local connections to diversified substitute vertices below the reference slice, while top-to-bottom bypass edges provide short routes that avoid the damaged slice. Algorithm 1 gives the complete construction. The resulting graphs $\widehat{H} \in \mathrm{Hydra}(n, k)$ have the following properties.

(i) *Short-path resilience.* For every $k \ge 3$ and $n \ge k^k$, $\widehat{H}$ is k-step k-connected. Consequently, after the failure of any $k-1$ intermediate vertices, every surviving pair is still joined by a path of length at most k; see Theorem 5.18(i).

(ii) *Decentralization at the optimal degree scale.* The degrees of $\widehat{H}$ satisfy

$$M_H - 1 \le \delta(\widehat{H}) \le \Delta(\widehat{H}) \le (k+1)\lfloor n^{1/k} \rfloor.$$

Thus, for fixed k, Hydra attains the maximum-degree scale imposed by the Moore bound, while all degrees remain of the same order. The connectivity burden is therefore distributed throughout the network rather than concentrated on a small number of hubs; see Corollary 5.16 and Theorem 5.18(iii).

(iii) *Hyperconnectivity.* The same graph is $(k+1)$-step $(M_H - 1)$-connected. Allowing one additional hop therefore raises the guaranteed number of independent routes between every pair from k to a degree-scale quantity of order $kn^{1/k}$; see Theorem 5.18(ii).

(iv) *Asymptotic edge optimality.* Let

$$e^\star(n; k+1, M_H - 1) := \min\big\{\|G\| : |G| = n,\ G \text{ is } (k+1)\text{-step } (M_H-1)\text{-connected}\big\}.$$

Theorem 5.19 gives

$$\frac{n(M_H-1)}{2} \le e^\star(n; k+1, M_H-1) \le \|\widehat{H}\| \le \frac{n(M_H+k)}{2}.$$

Consequently,

$$1 \le \frac{\|\widehat{H}\|}{e^\star(n; k+1, M_H-1)} \le 1 + \frac{k+1}{M_H-1} \longrightarrow 1$$

as $n \to \infty$ for fixed k. Hence Hydra is asymptotically edge-optimal for its stronger hyperconnectivity guarantee.

(v) *Dynamic maintainability.* Following a deletion, Algorithm 3 produces a graph in Hydra$(n-1, k)$ whenever $n - 1 \ge k^k$. Algorithm 4 similarly incorporates a new vertex into a graph in Hydra$(n+1, k)$. Theorem 5.20 proves that these updates keep the graph inside the Hydra family. Iterating them therefore restores properties (i)–(iv) after any sequence of arrivals and departures for which the updated network sizes remain feasible.

These properties motivate the name *Hyperconnected Decentralized Rewiring Architecture*, or *Hydra*. The acronym also evokes the Lernaean Hydra of Greek mythology, which regenerated its severed heads. Analogously, after a vertex is removed, the surviving network forms new links and recovers the architecture's defining guarantees.

The hop budget k remains a design parameter. For fixed k, Hydra has degree of order $kn^{1/k}$ and an edge count of order $kn^{1+1/k}$. Increasing k permits longer routes but tends to reduce the number of links that must be maintained. It also changes both the number and the length of the independent routes supplied by the architecture. The preferred value of k therefore depends on the relative importance of maintenance, usage, and reliability.

If links have a common maintenance cost and there is no separate cost associated with route length or failed interactions, the designer chooses k solely to minimize the Hydra edge count. Let

$$K_n := \max\{k \in \mathbb{N} : k \ge 3,\ k^k \le n\}$$

be the largest feasible dimension. Proposition 6.1 shows that, for all sufficiently large n, K_n is the unique edge-minimizing dimension within the Hydra family and

$$K_n \sim \frac{\log n}{\log\log n}, \qquad \|\widehat{H}_n\| = \left(\frac{1}{2} + o(1)\right) nK_n^2 = \Theta\left(n\left(\frac{\log n}{\log\log n}\right)^2\right).$$

Thus maintenance alone favours the largest feasible value of k, while producing a near-linear edge count $n^{1+o(1)}$.

Reliability goes in the opposite direction. Suppose that every edge fails independently with probability p. Using only k-step k-connectivity gives the interaction-failure bound

$$\pi_k(p) = \Big[1 - (1-p)^k\Big]^k.$$

Increasing k now has two opposing effects: it provides more independent routes but permits each route to be longer. Consequently, $\pi_k(p)$ may increase or decrease with k, depending on p. Hyperconnectivity reveals a much stronger conclusion. By allowing just one additional step, Hydra provides at least

$$m_{n,k} = k\big(\lfloor n^{1/k} \rfloor - 1\big) - 1$$

edge-disjoint routes between every pair, leading to the bound

$$\bar{\pi}_{n,k}(p) = \left[1 - (1-p)^{k+1}\right]^{m_{n,k}}.$$

Proposition 6.2 shows that this bound is nonincreasing in n and strictly increasing in k. Under this hyperconnectivity-based reliability criterion, the smallest admissible dimension $k = 3$ is therefore optimal. The magnitude of the improvement can be substantial. As Tables 3 and 4 illustrate, when $n = 1{,}000$, $k = 3$, and $p = 1/3$, using only three paths of length at most three gives $\pi_3(1/3) = 34.85\%$, whereas allowing paths of length at most four reveals at least 26 independent routes and reduces the bound to $\bar{\pi}_{1000,3}(1/3) = 0.33\%$.

Maintenance and reliability therefore point toward opposite ends of the feasible range. To study their interaction with usage costs, Section 6.3 supposes that aggregate activity satisfies $T_n = \lambda n^{\alpha}$ and introduces the reduced-form objective

$$C_{\alpha}(k; n, \lambda, u, v, p) = \frac{n}{2} k n^{1/k} + uk\lambda n^{\alpha} + v\lambda n^{\alpha} \bar{\pi}_{n,k}(p),$$

where u is the cost of one hop and v is the loss generated by a failed interaction. Theorem 6.4 shows that the resulting phase diagram contains three economic regimes. When $\alpha = 1$, activity per vertex remains bounded and maintenance dominates, so $k = K_n$ is eventually optimal. When $1 < \alpha < 4/3$, the optimum is an intermediate fixed dimension: at the critical values $\alpha = 1 + 1/j$, the relative maintenance and usage cost λu determines which of the adjacent dimensions j and $j + 1$ is selected; away from the critical exponents it is $k_n^{\star} = \lceil 1/(\alpha - 1) \rceil$. When $\alpha > 4/3$, usage dominates, so the shortest-route architecture $k = 3$ is eventually optimal.

Although failed interactions enter the joint objective explicitly, hyperconnectivity makes their contribution asymptotically negligible relative to maintenance and usage costs. Reliability can nevertheless remain economically important at finite network sizes. Section 6.4 illustrates this point for the Lightning Network. Applying Hydra to every public node would exceed the channel budgets of ordinary users, so we instead propose a Hydra routing layer maintained by professional routing and liquidity providers. Ordinary users connect to a small number of these providers, while the choices of the routing-layer size N, dimension k, and access degree d jointly determine channel requirements and reliability.

These results move the paper beyond a purely topological feasibility question. Hydra is a first step toward an economic theory of decentralized connectivity, but the framework also identifies how maintenance, activity, and reliability select among the architectures in the family and how the resulting design can be adapted to an operational network.

The remainder of the paper is organized as follows. Section 2 gives an informal geometric account of Hydra, beginning with the Hamming benchmark and explaining the repair, bypass, and update mechanisms. Section 3 interprets the model as a connectivity-budget problem for blockchain infrastructure and relates Hydra to validator and full-node gossip, transaction relay, and DeFi market structure. Section 4 compares our approach to the literatures on extremal graph theory, interconnection networks, self-healing networks, and survivable network design. Section 5 presents the formal construction and establishes its connectivity, degree, edge-optimality, and update properties. Section 6 develops the economic interpretation, identifies the maintenance-minimizing dimension, derives reliability bounds under independent edge failures, studies the joint choice of k under power-law activity, and applies the resulting framework to a Hydra routing layer for the Lightning Network. Section 7 concludes. The appendices collect notation, comparisons with related extremal problems, an integer linear programming formulation, the comparison with expander graphs, and all proofs.

2 Hydra

This section gives an informal overview of *Hydra*. The formal definitions, proofs, and optimality statements are deferred to Section 5. We begin with an idealized benchmark and then explain

how Hydra preserves its main features for arbitrary network sizes and under the insertion or deletion of vertices.

The easiest case to visualize is $k = 3$ and $n = q^3$. Place the vertices at the points $\{1, \dots, q\}^3$ of a $q \times q \times q$ cube. Two distinct vertices are connected whenever they agree in two coordinates. In other words, any two points lying on the same line in the x-, y-, or z-direction are connected. The resulting graph is the Hamming graph $H(q, q, q)$. Its coordinate structure creates short routes in a transparent way. For example, one can travel from $x = (x_1, x_2, x_3)$ to $y = (y_1, y_2, y_3)$ by changing the three coordinates successively:

$$(x_1, x_2, x_3) \longrightarrow (y_1, x_2, x_3) \longrightarrow (y_1, y_2, x_3) \longrightarrow (y_1, y_2, y_3).$$

Changing the coordinates in different orders gives multiple short paths. When some coordinates already agree, detours through unused coordinate values provide alternative routes.

Each vertex has $q - 1$ neighbors in each of the x-, y-, and z-directions. Hence every vertex has the same degree $M_H = 3(q - 1)$. Between every pair of vertices, there are at least three independent paths of length at most three. If one additional step is allowed, there are M_H independent paths of length at most four. Thus a single additional hop increases the guaranteed number of independent routes from the dimension to the full degree of the graph.

Nothing essential is special to three dimensions. If $n = q^k$, the same construction on $\{1, \dots, q\}^k$ gives the Hamming graph $H(q, \dots, q)$, whose degree is $M_H = k(q - 1)$. It is k-step k-connected and $(k + 1)$-step M_H-connected. The latter property is also edge-optimal:

$$\|H(q, \dots, q)\| = \frac{nM_H}{2} = \min\Big\{\|G\| : |G| = n,\ G \text{ is } (k+1)\text{-step } M_H\text{-connected}\Big\}.$$

Indeed, M_H independent paths from each vertex require that every vertex have at least M_H neighbors. The handshaking lemma therefore gives the lower bound $nM_H/2$, which the M_H-regular Hamming graph attains exactly. Consequently $H(q, \dots, q)$ is resilient, decentralized, and edge-optimal.

The perfect-power case is an important benchmark, but it is also exceptional: in practice, the number of participants in an operational network changes over time. Hydra allows for an arbitrary number n by starting from a near-balanced Hamming graph that has at least n vertices. More precisely, we choose side lengths $\lfloor n^{1/k} \rfloor \le q_1, \dots, q_k \le \lfloor n^{1/k} \rfloor + 1$, as specified in Proposition 5.13, that are as balanced as possible, with $Q := \prod_{\ell=1}^{k} q_\ell \ge n$, while keeping their sum small. The resulting ambient Hamming graph $H(q_1, \dots, q_k)$ has Q vertices and degree $M_H := \sum_{\ell=1}^{k}(q_\ell - 1)$ of order $n^{1/k}$; so keeping the side lengths balanced also keeps the number of connections at each vertex low. We then delete $t := Q - n$ vertices from the top slice $S := \{x : x_k = q_k\}$. The balanced choice of side lengths guarantees that this slice contains enough vertices to do so. Deletion alone, however, destroys some of the short paths available in the full Hamming graph.

Hydra compensates for these lost routes using two types of additional links, illustrated in Figure 1. The first is an *anchor repair*. In the three-dimensional picture, let

$$v = (v_1, v_2, q_3) \in D$$

be one of the vertices deleted from the top slice. For every surviving neighbor of v in the first coordinate direction, Hydra adds an edge connecting that neighbor to the anchor

$$(v_1, v_2, q_3 - 1).$$

For every surviving neighbor of v in the second coordinate direction, Hydra instead adds an edge connecting that neighbor to the distinct anchor

$$(v_1, v_2, q_3 - 2).$$

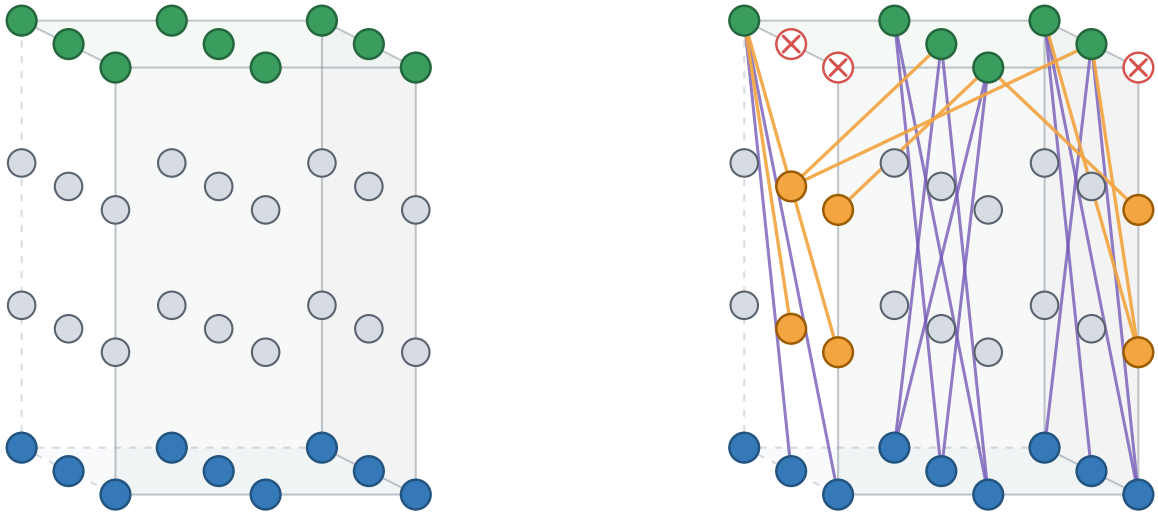


Figure 1: The ambient Hamming graph $H(3,3,4)$ and the corresponding construction Hydra$(33,3)$. Left: the ambient graph, represented by its four coordinate layers. Right: the Hydra graph obtained by deleting three top-slice vertices and adding anchor-repair edges in orange and top-to-bottom bypass edges in purple. Top-slice vertices are green, bottom-slice vertices are blue, and the remaining vertices are grey; red crosses mark the deleted coordinates. The faint outlines indicate the ambient coordinate boxes, while ordinary Hamming edges are omitted for visual clarity.

Thus Hydra transfers the local connections lost through the deletion of v to two distinct vertices below the top face. In k dimensions, the same rule assigns a different anchor depth to each of the $k-1$ coordinate directions within the top slice.

The second type is a *bypass link*. Each surviving top-slice vertex is connected to a collection of vertices in the bottom slice. These top-to-bottom links provide short routes that avoid the damaged top slice altogether. Algorithm 1 gives the complete k-dimensional construction. The resulting graph is called Hydra, short for Hyperconnected, Decentralized, Rewiring Architecture. Relative to the perfect-power Hamming benchmark, it retains the following properties.

Short-path resilience and hyperconnectivity. Hydra is k-step k-connected; see Theorem 5.18(i). Allowing one additional step reveals a substantially stronger property. Hydra is $(k+1)$-step (M_H-1)-connected. Hence the number of guaranteed independent paths increases from k to M_H-1, which is of order $n^{1/k}$. This is the hyperconnectivity property established in Theorem 5.18(ii).

Edge economy. The stronger connectivity guarantee is obtained at close to the smallest possible number of edges. Theorem 5.19 gives the following comparison:

$$\begin{aligned}\frac{n(M_H-1)}{2} &\le \min\Big\{\|G\| : |G|=n,\ G \text{ is } (k+1)\text{-step } (M_H-1)\text{-connected}\Big\}\\ &\le \|\widehat{H}\| \le \frac{n(M_H+k)}{2}.\end{aligned}$$

The first inequality is the universal degree lower bound: a graph providing M_H-1 independent paths between every pair must have minimum degree at least M_H-1. The final inequality shows that Hydra lies within at most $n(k+1)/2$ edges of this lower bound.

Algorithm 1 Hydra construction

Require: Integers $k \geq 3$ and $n \geq k^k$
Ensure: A graph $\widehat{H}$ on n vertices

Choose side lengths $q_1, \ldots, q_k$ as balanced as possible for (n, k) using Proposition 5.13.
Set $Q := \prod_{\ell=1}^{k} q_\ell$ and $t := Q - n$.
Initialize $\widehat{H} := H(q_1, \ldots, q_k)$ on the grid $V := \{(x_1, \ldots, x_k) : \ x_\ell \in \{1, \ldots, q_\ell\}\}$.
Let $S := \{x \in V : \ x_k = q_k\}$ be the top slice, and choose any set $D \subset S$ with $|D| = t$.
Delete the vertices in D.
for each deleted vertex $v = (v_1, \ldots, v_{k-1}, q_k) \in D$ **do**
    for each axis $a \in \{1, \ldots, k-1\}$ **do**
        Define the anchor $v^{(-a)} := (v_1, \ldots, v_{k-1}, q_k - a)$.
        Let $N_a(v) := \{y \in S : \ y_\ell = v_\ell \ \forall \ell \neq a, \ y_a \neq v_a\}$ be the neighbors of v in the top slice along axis a.
        for all $y \in N_a(v) \setminus D$ **do**
            Add the edge $y - v^{(-a)}$.
        end for
    end for
end for
for each surviving top-slice vertex $x = (x_1, \ldots, x_{k-1}, q_k) \in S \setminus D$ **do**
    for all $x_1' \in \{1, \ldots, q_1\} \setminus \{x_1\}$ **do**
        Add the edge $x - (x_1', x_2, \ldots, x_{k-1}, 1)$.
    end for
end for

Writing $e^\star(n; k+1, M_H - 1)$ for this minimum, we obtain

$$1 \leq \frac{\|\widehat{H}\|}{e^\star(n; k+1, M_H - 1)} \leq 1 + \frac{k+1}{M_H - 1}.$$

Since M_H is of order $n^{1/k}$, the upper bound converges to one as $n \to \infty$. Thus the additional hop raises the guaranteed path multiplicity from k to $M_H - 1$, while Hydra's edge count remains asymptotically minimal.

Decentralization. Hydra maintains the degree scale of the near-balanced Hamming graph without creating high-degree hubs. The degree bounds in Theorem 5.18(iii) give

$$\delta(\widehat{H}) \geq M_H - 1 \geq k\big(\lfloor n^{1/k} \rfloor - 1\big) - 1, \qquad \Delta(\widehat{H}) \leq (k+1)\lfloor n^{1/k} \rfloor.$$

Therefore, the burden of maintaining connections is distributed across the network rather than concentrated on a small number of vertices.

Self-healing. Beyond these static guarantees, Hydra admits explicit update rules as vertices enter or leave the network. Following a deletion, the coordinate assignment is rearranged if necessary so that the new vacancy lies in the reference top slice. The deletion set and its associated repairs are then updated, or, when the entire top slice has been removed, the ambient grid is contracted to the next balanced size. An insertion reverses this procedure: the new vertex fills an available top-slice position whenever one exists, while a full ambient grid is expanded using the next balanced set of side lengths.

The formal deletion and insertion procedures are given in Algorithms 3 and 4. Theorem 5.20 shows that both operations keep the network inside the Hydra family. Consequently, Hydra is not merely a static graph construction, but a dynamically maintainable network architecture

whose short-path resilience, hyperconnectivity, decentralized degree scale, and near-optimal edge count persist as participants enter and leave.

Table 1 summarizes Hydra's key features, the mechanisms through which they are obtained, and the corresponding formal results.

Feature	Construction mechanism	Formal result
Short-path resilience	Coordinate routes and anchor repairs	Theorem 5.18(i)
Hyperconnectivity	One additional hop using repairs and bypasses	Theorem 5.18(ii)
Decentralization	Balanced side lengths and distributed repairs	Theorem 5.18(iii)
Near edge-optimality	Hamming backbone with sparse repairs	Theorem 5.19
Self-healing	Top-slice updates and balanced grid resizing	Theorem 5.20

Table 1: Hydra's main features, their construction mechanisms, and the corresponding formal results.

3 Connectivity Budgets in Blockchain Infrastructure

DeFi and digital-currency protocols achieve decentralization not only through cryptography and incentives, but also through connectivity. A validator that cannot hear blocks on time, a full node that receives transactions through a small set of intermediaries, or a data-availability participant that cannot obtain enough shares before a deadline may be economically decentralized on paper but operationally dependent on a narrow communication layer. Connectivity is therefore a protocol resource. It requires maintained peers, upload capacity, routing reliability, monitoring, protection against denial-of-service attacks, and rules for replacing failed neighbors. The graph $G = (V, E)$ abstracts this layer: vertices are validators, full nodes, relays, data-availability participants, or channel-network participants; edges are standing communication, bandwidth, routing, or liquidity commitments; $\Delta(G)$ measures the largest per-node fanout burden; and $\|G\|$ measures the aggregate connectivity budget.

The paper's primitives translate directly into blockchain infrastructure. The hop budget k is a latency or timeliness constraint: blocks, attestations, transactions, or data chunks must traverse the overlay within a small number of relay steps. The redundancy parameter m is a resilience constraint: propagation should survive censorship, congestion, eclipse attempts, targeted node failures, or dropout by using several internally vertex-disjoint short routes. The maximum degree $\Delta(G)$ is the per-node burden of decentralization, because high-degree nodes carry more bandwidth, operational exposure, and attack surface. The edge count $\|G\|$ is the aggregate cost of sustaining the overlay. Finally, self-healing corresponds to peer rotation, reconnect logic, channel rebalancing, topology repair, or other mechanisms that restore connectivity after churn.

Two scaling relations summarize the cost of this resource. Let $\Delta^\star(n; k, m)$ denote the smallest maximum degree among n-vertex graphs that are k-step m-connected. For fixed $k \geq 3$ and $2 \leq m \leq k$, Corollary 5.16 and Theorem 5.18 imply

$$\Delta^\star(n; k, m) = \Theta\Big(n^{1/k}\Big). \tag{3.1}$$

Thus, if the protocol fixes a strict hop budget and insists on hub-free short-path redundancy, the worst-case peer obligation cannot remain constant as the network grows. Alternatively, if the designer allows the hop budget to grow and minimizes the maintenance cost within the Hydra family, Proposition 6.1 gives

$$\|\widehat{H}_n\| = \left(\frac{1}{2} + o(1)\right) nK_n^2 = \Theta\left(n\left(\frac{\log n}{\log\log n}\right)^2\right), \qquad K_n \sim \frac{\log n}{\log\log n}. \tag{3.2}$$

Model object	Blockchain interpretation	Economic margin
Vertex $v \in V$	validator, beacon node, full node, relay, DA participant, payment-channel node	entity that supplies routing or verification capacity
Edge $uv \in E$	peering session, relay relation, upload commitment, routing channel, liquidity channel	recurring bandwidth, operational, security, or collateral cost
Hop budget k	propagation deadline measured in overlay hops	settlement speed, timely consensus participation, or data-retrieval deadline
Redundancy m	number of independent short routes	tolerance to censorship, eclipse attacks, congestion, targeted failures, or dropout
Degree $\Delta(G)$	largest peer set, fanout, upload obligation, or channel exposure	decentralization of operational burden and attack surface
Edge count $\|G\|$	aggregate peering, bandwidth, routing, or liquidity budget	total connectivity cost paid by the protocol community
Self-healing	peer replacement, topology repair, rebalancing, reconnect rules	restoration of robustness after churn or attack

Table 2: Interpreting the paper's graph primitives as blockchain-infrastructure objects.

Equations (3.1) and (3.2) are not implementation prescriptions. They are accounting constraints: a protocol that wants fast, redundant, hub-free propagation must either fund the corresponding connectivity budget, relax latency or redundancy, or tolerate a more centralized relay structure.

3.1 Validator and Full-Node Gossip Overlays

Ethereum's consensus-layer communication illustrates the distinction between validators and the connectivity substrate. A validator client typically signs duties and submits signed objects to one or more beacon nodes; the beacon nodes participate in the peer-to-peer network. Blocks, attestations, aggregate attestations, and related consensus messages are then propagated through GossipSub. For a topic such as beacon blocks or aggregate attestations, the resulting overlay is well approximated by a dynamic bounded-degree random-regular-like graph: each node maintains a small topic mesh, validates newly received messages, forwards unseen valid messages to mesh peers, and suppresses already-seen duplicates. The Ethereum consensus specifications set the GossipSub target mesh degree at $D = 8$, with low and high watermarks $D_{\text{low}} = 6$ and $D_{\text{high}} = 12$, and use $D_{\text{lazy}} = 6$ for metadata gossip; raw attestations are disseminated through attestation subnets, while beacon blocks and aggregate attestations are propagated on broader topics (Ethereum Foundation, 2026; Vyzovitis et al., 2020).

This design is neither a complete graph nor a tree. A complete graph would minimize hop distance but would impose quadratic peering cost. A tree would minimize duplicate transmissions but would be fragile: removing a single internal relay can disconnect an entire subtree from timely propagation. GossipSub instead uses sparse mesh redundancy. It accepts some duplicate traffic in exchange for low latency, robustness to churn, and resistance to local peer failures. In the language of this paper, Ethereum fixes a small per-topic degree and relies on a probabilistic, dynamic, peer-scored overlay rather than requiring a deterministic k-step m-connected graph for every pair of nodes.

Hydra should therefore not be read as a literal description of Ethereum networking. Its role

is a benchmark. If one imposes the stronger deterministic requirement that every pair of nodes have m internally vertex-disjoint routes of length at most k, with no privileged relay nodes, then (3.1) states the minimum order of per-node connectivity that must be supplied. Conversely, if a system keeps degree close to a constant, as a bounded-degree gossip mesh does, then some combination of the hop budget, the pairwise guarantee, or the redundancy requirement must be weakened. This is the economic content of the graph bound: constant fanout is cheap, but it cannot simultaneously deliver fixed-hop deterministic redundancy for an arbitrarily large hub-free network.

Bitcoin mempool relay is the same economic problem in a different networking architecture. Transaction propagation runs over a bounded peer graph that is closer to a fixed-outdegree graph with heterogeneous indegree than to an Ethereum-style per-topic mesh. Nodes announce transactions and peers request the ones they are missing, rather than sending every full transaction body immediately to every neighbor. Naumenko et al. (2019) make the connectivity trade-off explicit: more peer connections improve security and robustness, but traditional transaction relay makes bandwidth grow with connectivity; Erlay uses set reconciliation to reduce duplicate announcement costs, allowing higher connectivity at lower bandwidth cost. In our terminology, Erlay is a mechanism for lowering the price of an edge in the transaction-relay graph, thereby making a larger and more robust connectivity budget feasible.

The Hamming structure behind Hydra also clarifies the difference between duplicate-free dissemination and robust dissemination. In an ideal Hamming graph, a coordinate-ordered broadcast can send a message along a spanning tree and reach all n vertices with $n-1$ transmissions. This is bandwidth-optimal but fragile, because a failed tree edge can silence a whole branch unless repair traffic is added. By contrast, full flooding on a d-regular overlay, with nodes forwarding a newly seen message to all neighbors except the one from which it was first received, uses $n(d-1)+1$ transmissions and creates duplicates, but gives several chances for a message to arrive through different intermediaries. Hydra uses Hamming-style coordinates for a different purpose: to characterize and construct low-cost hub-free short-path redundancy, together with anchor-based repair after deletions. It is a cost benchmark and repair template, not a replacement for protocol-level duplicate suppression, peer scoring, or message-validation rules.

3.2 Edges as Costly Commitments in DeFi Market Structure

The same interpretation applies beyond block and data propagation. In payment-channel networks, an edge is not merely a communication link: it is a bilateral liquidity commitment. Degree corresponds to locked capital, monitoring burden, and routing exposure; self-healing corresponds to rebalancing, closing, reopening, or rerouting channels after liquidity shocks. The graph-theoretic question is again whether the network can provide many short routes without concentrating routing power in a few highly connected intermediaries.

For decentralized exchanges, the natural connection is not the invariant curve of an automated market maker, but the infrastructure layer underneath market access. Zoican and Zoican (2019) show that infrastructure asymmetry can fragment distributed exchanges by creating speed advantages across regions, while Lehar et al. (2023) study how gas costs shape liquidity supply across decentralized venues. Our framework is complementary: it asks how much hub-free connectivity must be maintained before agents in different regions, venues, or protocol layers can interact without relying on privileged relays. Thus an edge can stand for a communication route, a routing/liquidity channel, or an operational commitment that makes market access timely and resilient.

Protocol-facing implications. The framework yields four implications for blockchain design. First, fixed-hop deterministic redundancy is incompatible with constant degree at arbitrary scale: a hard fanout cap D forces the feasible hop budget to grow at least logarithmically in

n, while a fixed hop budget forces degree to grow as in (3.1). Second, minimizing aggregate links does not reduce the connectivity problem to a tree; robust hub-free connectivity remains superlinear in the maintenance-cost benchmark (3.2). Third, duplicate-free propagation and robust propagation solve different problems: the former economizes on transmissions in a failure-free state, while the latter buys alternative routes under latency, churn, and adversarial disruption. Fourth, repair must be topology-aware. Reconnecting to arbitrary available peers may restore degree but not internally vertex-disjoint short paths; anchor-style repair is valuable because it restores the relevant robustness property rather than merely replacing lost edges.

This section positions the paper within blockchain economics by shifting attention from consensus incentives to the connectivity layer that consensus presupposes. Biais et al. (2019) and Saleh (2021) study how ledger protocols can be sustained by incentives and consensus design; Amoussou-Guenou et al. (2019a) analyze rational and Byzantine behavior in committee-based blockchains; and Amoussou-Guenou et al. (2019b) show that fair rewards in committee-based blockchains depend on communication conditions. Our contribution is to price those communication conditions as graph-theoretic constraints. This paper isolates a prior resource-allocation problem: how many connections, at what maximum per-node burden, are necessary to make short redundant hub-free communication feasible, and how can that structure be repaired after disruption?

4 Literature Review

Designing large-scale networked systems that guarantee connectivity under degree and cost constraints, while maintaining resilience to adversarial attacks and failures, is a central question that cuts across several strands of literature in mathematics, economics, computer science, and operations research. Despite extensive prior work, existing approaches typically address only subsets of the design objectives considered in this paper. In particular, the simultaneous achievement of bounded decentralized degree, guaranteed multipath connectivity, scalability to arbitrary network sizes, and dynamic self-healing under node removals remains largely unexplored in the literature.

The literature in graph theory primarily focuses on connectivity guarantees through extremal theory and algebraic constructions that minimize the number of edges while satisfying connectivity constraints. These results typically rely on specialized constructions applicable only to restricted network sizes, incorporate centralized network topology, and focus primarily on small-diameter or two-hop connectivity regimes. Classical results characterize minimum edge requirements for k-connectivity and path existence Menger (1927); Bollobás (2004), while subsequent work derives degree-based sufficient conditions for the existence of multiple internally disjoint paths and linkage structures Faudree et al. (1989, 1992); Faudree and Tuza (1995). In all of these constructions, at least one node is forced to have a degree that grows linearly with network size, resulting in a centralized topology. Constructing decentralized networks that balance degree–diameter trade-offs in large networks remains a fundamental challenge in network design, as highlighted by Miller and Širáň (2013), which quantifies the limits on network size given constraints on node degree and diameter.

Despite these advances, general multi-hop path multiplicity guarantees with bounded decentralized degrees remain limited. Furthermore, the literature typically considers static network structures, and constructive self-healing mechanisms that dynamically restore connectivity after node removals are largely absent. Consequently, existing frameworks provide important theoretical benchmarks but do not directly address the combined challenges of scalability, decentralization, and adaptive resilience in large operational networks.

The computer science literature extensively analyzes the connectivity and routing performance of specific structured network topologies, such as hypercube networks and their generalizations Bhuyan and Agrawal (1984); Harary et al. (1988), as well as flip-tree graphs Meyer

(2002), and has introduced algorithms for associated combinatorial optimization problems Lu et al. (2022). Most studies emphasize performance evaluation rather than structural optimality under degree constraints. Furthermore, network topology is typically assumed to be static, with failures handled through replication or rerouting rather than through topological repair. More recent work on self-healing networks proposes mechanisms that locally rewire connections following adversarial node deletions to preserve global properties such as connectivity Saia and Trehan (2008); Hayes et al. (2009); Pandurangan et al. (2016). However, these approaches generally do not explicitly optimize degree distributions or multipath connectivity guarantees, which are central to operational network design.

In operations research, network robustness has been studied primarily through survivable network design and reliability optimization models, which aim to construct minimum-cost networks that satisfy pairwise connectivity requirements while accounting for edge connectivity and stochastic failures Magnanti and Wong (1984); Myung et al. (1999); Kerivin and Mahjoub (2005). These optimization approaches provide explicit cost-connectivity trade-offs, but they typically assume static network topologies and do not incorporate decentralized self-healing mechanisms. Furthermore, due to the lack of analytic characterizations, solving these optimization models at scale is computationally challenging, often requiring relaxations or heuristics that limit their applicability to very large networks.

5 Main Results

Throughout, all graphs are finite, simple, and undirected. We follow the standard graph-theoretic terminology of Diestel (2025); all notation is collected in Appendix A.

A *graph* is an ordered pair $G = (V, E)$, where $V \neq \emptyset$ and $E \subset \{\{x, y\} : x, y \in V,\ x \neq y\}$. The elements of V and E are called the *vertices* and *edges* of G, respectively. The number of vertices is the *order* of G, denoted by $|G|$, while the number of edges is its *size*, denoted by $\|G\|$.

A *walk* of length $t \geq 1$ from $x \in V$ to $y \in V$ is a sequence $(v_0, \ldots, v_t)$ such that $v_0 = x$, $v_t = y$, and $\{v_{\ell-1}, v_\ell\} \in E$ for every $\ell \in \{1, \ldots, t\}$. An $x-y$ *path* is a walk from x to y whose vertices are all distinct. Two $x-y$ paths are *independent*, or *internally vertex-disjoint*, if they share no vertices other than x and y. A collection of paths is independent if its members are pairwise independent.

This leads to our central notion of *k-step m-connectedness*, which requires every pair of vertices to be joined by multiple short, independent paths, thereby combining resilience with efficiency. Connections with related notions are discussed in Appendix B.1.

Definition 5.1. For integers $k, m \geq 1$, a graph $G = (V, E)$ is *k-step m-connected* if, for every pair of distinct vertices $x, y \in V$, there exist at least m independent $x-y$ paths, each of length at most k.

This property was introduced by Faudree et al. (1987) under the shorthand $P_{k,m}$. Subsequent work developed sufficient conditions for a graph to satisfy $P_{k,m}$; see Faudree et al. (1989, 1992); Faudree and Tuza (1995). Our focus, by contrast, is the corresponding extremal design problem: for fixed n, k, and m, how small can the maximum degree be, and, among graphs attaining this smallest degree, how few edges are needed? More precisely, we study

$$\min\Big\{\big(\Delta(G), \|G\|\big) : |G| = n,\ G \text{ is } k\text{-step } m\text{-connected}\Big\}, \tag{5.1}$$

where the minimum is taken in lexicographic order. Thus, decentralization, as measured by the maximum degree $\Delta(G)$, is the primary objective; among degree-optimal graphs, edge economy $\|G\|$ is secondary.

Appendix B.2 compares (5.1) with several related extremal problems that instead prioritize the number of edges. A recurring feature of these sparsity-first benchmarks is that their extremal constructions can contain *hubs*. This is why we place decentralization first.

Remark 5.2. The lexicographic optimization problem (5.1) admits a direct integer linear programming formulation. Binary variables select the edges, degree constraints control the maximum degree, and auxiliary path variables enforce the existence of m internally vertex-disjoint paths of length at most k between every pair of distinct vertices. Although conceptually straightforward, the formulation grows rapidly with n, making it most useful for validation at small orders. For completeness, Appendix C gives the full formulation.

5.1 Minimum Possible Maximum Degree of k-Step m-Connected Graphs

To address the primary objective in (5.1), we first determine the smallest possible maximum degree of an n-vertex graph that is k-step m-connected. For integers $k \geq 2$, $m \geq 2$, and $n \geq m+1$, define

$$\Delta^\star(n;k,m) \coloneqq \min\big\{\Delta(G) : |G| = n \text{ and } G \text{ is } k\text{-step } m\text{-connected}\big\}.$$

The restriction $n \geq m+1$ ensures that the feasible class is nonempty: indeed, the complete graph K_n is 2-step $(n-1)$-connected.

For fixed $k \geq 2$ and $2 \leq m \leq k$, we will prove that, for all sufficiently large n,

$$(n-1)^{1/k} \leq \Delta^\star(n;k,m) \leq (k+1)\lfloor n^{1/k} \rfloor. \tag{5.2}$$

Consequently, $\Delta^\star(n;k,m) = \Theta(n^{1/k})$.

The lower bound follows directly from the Moore bound. Indeed, every k-step m-connected graph satisfies $\Delta(G) \geq \delta(G) \geq m \geq 2$ under the present assumptions. Now let G have maximum degree Δ and diameter at most k. The Moore bound gives

$$|G| \leq 1 + \Delta \sum_{i=0}^{k-1} (\Delta - 1)^i \leq 1 + \Delta^k.$$

Since every k-step m-connected graph has diameter at most k, it follows that

$$\Delta^\star(n;k,m) \geq (n-1)^{1/k}. \tag{5.3}$$

This estimate uses only the diameter requirement, rather than the full force of k-step m-connectedness. Nevertheless, it identifies $n^{1/k}$ as an unavoidable maximum-degree scale.

The matching upper bound is constructive. The Hydra construction developed below produces, for every sufficiently large n, an n-vertex graph that is k-step k-connected, and hence k-step m-connected for every $2 \leq m \leq k$, with

$$\Delta(G) \leq (k+1)\lfloor n^{1/k} \rfloor.$$

It therefore attains, up to a constant factor depending only on k, the smallest maximum-degree scale permitted by the Moore bound.

5.1.1 The Case $k = m = 2$: Near-Square Patched Rook Graphs

We begin with the case $k = m = 2$, which already exhibits the main features of the general problem. We introduce a family of graphs that are 2-step 2-connected and have maximum degree of order $\sqrt{n}$. The construction is based on the *Cartesian product* of complete graphs.

Definition 5.3. Let $G = (V_G, E_G)$ and $H = (V_H, E_H)$ be two graphs. Their *Cartesian product* $G \,\square\, H$ is the graph with vertex set $V(G \,\square\, H) = V_G \times V_H$ and edge set

$$E(G \,\square\, H) = \Big\{\{(g,h),(g',h')\} : \big(g = g' \text{ and } \{h,h'\} \in E_H\big) \text{ or } \big(h = h' \text{ and } \{g,g'\} \in E_G\big)\Big\}.$$

Definition 5.4. For integers $r, c \geq 2$, the $r \times c$ *rook graph* is

$$R_{r,c} \coloneqq K_r \,\square\, K_c,$$

the Cartesian product of the complete graphs K_r and K_c. Its vertices are pairs (i, j) with $i \in \{1, \ldots, r\}$ and $j \in \{1, \ldots, c\}$; two vertices $(i, j) \neq (i', j')$ are adjacent if $i = i'$ or $j = j'$.

The graph $R_{r,c}$ has rc vertices, each of degree $r + c - 2$, and therefore has $rc(r + c - 2)/2$ edges. Two nonadjacent vertices (i, j) and (i', j') have exactly two common neighbors, namely the coordinate crossings (i, j') and (i', j). Two adjacent vertices in the same row have $c - 2$ common neighbors, while two adjacent vertices in the same column have $r-2$ common neighbors. Hence, when $r, c \geq 3$, every nonadjacent pair is joined by two independent paths of length two, while every adjacent pair is joined by its common edge and at least one additional path of length two. Thus $R_{r,c}$ is 2-step 2-connected.

If $n = rc$ for integers $r, c \geq 3$, the rook graph $R_{r,c}$ is a 2-step 2-connected graph with maximum degree $r + c - 2$. Since this quantity is smallest when the two side lengths are close, one would ideally take $r \approx c \approx \sqrt{n}$. For general n, however, there need not be a suitably balanced exact factorization $n = rc$. This raises the following question: can we start from a near-square rook graph on at least n vertices, delete the excess vertices, and restore the lost short connections without exceeding the maximum degree of the ambient graph?

Set $r \coloneqq \lfloor \sqrt{n} \rfloor$ and $c \coloneqq \left\lceil \frac{n}{r} \right\rceil$. The resulting rectangle has at least n positions, its side lengths differ by at most two, and the number of excess positions is smaller than its shorter side. These elementary properties are collected below.

Proposition 5.5. *Let $n \geq 1$, set $r \coloneqq \lfloor \sqrt{n} \rfloor$ and $c \coloneqq \lceil n/r \rceil$. Then:*

(1) $rc \geq n$;

(2) $c \in \{r, r + 1, r + 2\}$;

(3) $r + c - 2 \leq 2\lceil \sqrt{n} \rceil - 2$;

(4) $0 \leq t \coloneqq rc - n < r$.

To obtain a graph on exactly n vertices, we delete the $t = rc-n$ excess vertices from the main diagonal of the ambient rook graph. Each deletion removes some of the two-step paths that pass through the deleted vertex. We restore these paths by matching the surviving vertices in the corresponding column with distinct surviving vertices in the corresponding row and adding patch edges between them.

Algorithm 2 formalizes this procedure. We write $\mathcal{R}_{\mathrm{ns}}(n)$ for the family of all graphs obtainable from the algorithm and call its members *near-square patched rook graphs*.

Remark 5.6. Algorithm 2 is well defined. Fix $\alpha \in \{1, \ldots, t\}$. After deleting (α, α), column α contains exactly $r-1$ surviving vertices, whereas row α contains $c-1 \geq r-1$ surviving vertices. For each surviving column vertex (β, α), the condition in Step 8 rules out at most one possible pairing, namely (α, β). Hence one can choose $r - 1$ distinct vertices in row α and match them bijectively with the surviving vertices in column α while respecting all forbidden pairs.

Figure 2 illustrates the construction for $n = 22$. The following theorem shows that, once $r \geq 4$, the patch edges restore all required short connections without raising the maximum degree above that of the ambient rook graph.

Theorem 5.7. *For every $n \geq 16$, every graph $G \in \mathcal{R}_{\mathrm{ns}}(n)$ is 2-step 2-connected and satisfies*

$$\Delta(G) \leq 2\lceil \sqrt{n} \rceil - 2.$$

Consequently, $\Delta^\star(n; 2, 2) = \Theta(\sqrt{n})$.

Algorithm 2 Near-square patched rook construction

Require: Integer $n \geq 4$
Ensure: A graph G on n vertices
1: Set $r := \lfloor \sqrt{n} \rfloor$, $c := \lceil \frac{n}{r} \rceil$ and $t := rc - n$.
2: Initialize $G := R_{r,c}$ on the grid $\{1, \ldots, r\} \times \{1, \ldots, c\}$.
3: **for** $\alpha = 1, \ldots, t$ **do**
4: Delete the vertex $d_\alpha := (\alpha, \alpha)$.
5: **end for**
6: **for** $\alpha = 1, \ldots, t$ **do**
7: Add $r - 1$ patch edges joining the $r - 1$ surviving vertices in column α to $r - 1$ distinct surviving vertices in row α.
8: Choose these patch edges so that for every $\beta \in \{1, \ldots, t\} \setminus \{\alpha\}$, the vertex (β, α) is not joined to (α, β).
9: **end for**

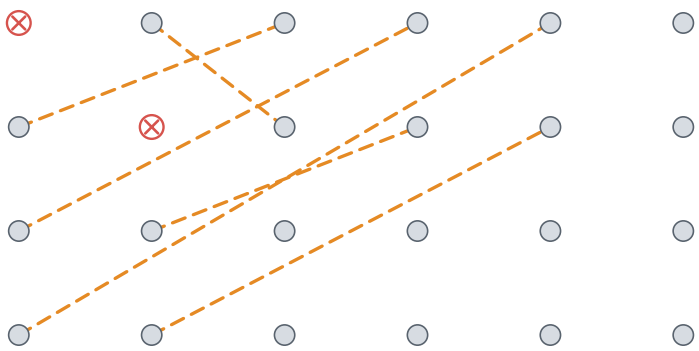


Figure 2: A near-square patched rook graph on $n = 22$ vertices, represented on a 4×6 grid. The red crosses mark the deleted coordinates $(1, 1)$ and $(2, 2)$, and the six dashed lines are patch edges. Ordinary rook edges are omitted for visual clarity: each row and each column induces a clique among its surviving vertices. Here $r = 4$, $c = 6$, and $\Delta = r + c - 2 = 8 = 2\lceil \sqrt{22} \rceil - 2$.

Remark 5.8. The restriction $n \geq 16$ in Theorem 5.7 ensures that $r = \lfloor \sqrt{n} \rfloor \geq 4$, which is needed for the patched-rook family to be 2-step 2-connected. For $n < 16$, the family $\mathcal{R}_{\mathrm{ns}}(n)$ may fail to have this property. Nevertheless, the bound

$$\Delta^\star(n; 2, 2) \leq 2\lceil \sqrt{n} \rceil - 2$$

still holds by explicit constructions for all $n \geq 5$. The remaining cases are trivial: $\Delta^\star(3; 2, 2) = 2$ and $\Delta^\star(4; 2, 2) = 3$.

5.1.2 Near-Balanced Patched Hamming Graphs

We now turn to the general case $k \geq 3$, and first treat the extremal resilience level $m = k$. The case $m < k$ will then follow immediately, since k-step k-connectivity implies k-step m-connectivity for every $m \leq k$. Building on the rook-graph construction from the case $k = m = 2$, we pass to higher dimensions via *Hamming graphs*.

Definition 5.9. For integers $k \geq 2$ and $q_1, \ldots, q_k \geq 2$, the *Hamming graph*

$$H(q_1, \ldots, q_k) := K_{q_1} \,\square\, K_{q_2} \,\square\, \cdots \,\square\, K_{q_k}$$

is the Cartesian product of the complete graphs $K_{q_1}, \ldots, K_{q_k}$. Its vertices are tuples $(i_1, \ldots, i_k)$ with $i_\ell \in \{1, \ldots, q_\ell\}$, and two vertices are adjacent if they differ in exactly one coordinate.

Hamming graphs are the natural higher-dimensional analogues of rook graphs. Every vertex has degree

$$\Delta = \sum_{\ell=1}^{k} (q_\ell - 1), \tag{5.4}$$

the graph has $\prod_{\ell=1}^{k} q_\ell$ vertices, and its diameter is k. More generally, the distance between two vertices is exactly the number of coordinates in which they differ. Shortest paths are therefore obtained by correcting the differing coordinates one at a time, with alternative geodesics arising from different orders of correction. When further independent routes are needed, the availability of a third value in each coordinate provides additional short detours. These two mechanisms yield the following result.

Proposition 5.10. *For integers $k \geq 2$ and $q_1, \ldots, q_k \geq 3$, the Hamming graph $H(q_1, \ldots, q_k)$ is k-step k-connected.*

Combining Proposition 5.10 with (5.3) and (5.4) already gives the correct order of growth when n is a perfect kth power.

Corollary 5.11. *If $n = q^k$ with $q \geq 3$ an integer and $k \geq 3$, then $\Delta^\star(n; k, k) = \Theta(n^{1/k})$. In particular, the Hamming graph $H := H(q, \ldots, q)$ is k-step k-connected with maximum degree*

$$\Delta(H) = k(q-1) = k(n^{1/k} - 1).$$

To treat arbitrary n, we choose an ambient Hamming graph with at least n vertices whose side lengths minimize the ambient degree while remaining as *balanced as possible*. We first formalize this choice and record its basic properties.

Definition 5.12. Let $n \geq 2^k$ and $k \geq 3$. Integers $q_1, \ldots, q_k \geq 2$ are called *as balanced as possible for (n, k)* if

(i) $\prod_{\ell=1}^{k} q_\ell \geq n$;

(ii) $|q_\ell - q_{\ell'}| \leq 1$ for all ℓ, ℓ' (all side lengths differ by at most one);

(iii) among all integer k-tuples $(p_1, \ldots, p_k)$ with $p_\ell \geq 2$ and $\prod_{\ell=1}^{k} p_\ell \geq n$, the sum $\sum_{\ell=1}^{k} q_\ell$ is minimal.

Proposition 5.13. *Let $n \geq 2^k$ and $k \geq 3$. Set $r := \lfloor n^{1/k} \rfloor$ and choose the smallest integer $s \in \{0, 1, \ldots, k\}$ such that*

$$Q_s := r^{k-s}(r+1)^s \geq n.$$

Define $q_1 = \cdots = q_{k-s} = r$ and $q_{k-s+1} = \cdots = q_k = r+1$. Then $(q_1, \ldots, q_k)$ is as balanced as possible in the sense of Definition 5.12. Moreover, with $Q := \prod_{\ell=1}^{k} q_\ell$ and $t := Q - n$, we have:

(1) *$Q \geq n$ and $q_\ell \in \{r, r+1\}$ for all ℓ;*

(2) *$k(r-1) \leq M_H \leq kr$, where $M_H := \sum_{\ell=1}^{k} (q_\ell - 1) = k(r-1) + s$;*

(3) *if $s = 0$ then $t = 0$, and if $s \geq 1$ then $0 \leq t < r^{k-s}(r+1)^{s-1}$.*

To obtain a graph on exactly n vertices, we apply the deletion-and-patching procedure introduced in Algorithm 1. Recall that

$$S := \{(x_1, \ldots, x_{k-1}, q_k) : x_\ell \in \{1, \ldots, q_\ell\}\}$$

denotes the top slice of the ambient near-balanced Hamming graph on $Q \geq n$ vertices. The construction deletes a set $D \subset S$ of $t = Q - n$ vertices and adds the two classes of patch edges specified in the algorithm. We write $\mathcal{H}_{\mathrm{nb}}(n, k)$ for the resulting family and call its members *near-balanced patched Hamming graphs*. Figure 1 illustrates the construction for $n = 33$ and $k = 3$.

Remark 5.14. Algorithm 1 is well defined. Set $r := \lfloor n^{1/k} \rfloor$, and let s be as in Proposition 5.13. If $t = 0$, then the deletion set is empty. If $t > 0$, then $s \geq 1$, and Proposition 5.13 gives

$$t < r^{k-s}(r+1)^{s-1} = \prod_{\ell=1}^{k-1} q_\ell = |S|.$$

Thus a deletion set $D \subset S$ of size t exists, and at least one top-slice vertex survives.

Moreover, $t > 0$ implies $q_k = r + 1$. Since $n \geq k^k$, we have $r \geq k$, and hence $q_k \geq k + 1$. Therefore, for every $a \in \{1, \ldots, k-1\}$, $2 \leq q_k - a \leq q_k - 1$, so every anchor $v^{(-a)}$ is a valid vertex of the ambient Hamming graph.

The construction is easiest to understand as follows. Deleting vertices from the top slice destroys some short independent paths through that slice. The anchor-repair edges replace these paths by detours through designated vertices below the top slice. The top-to-bottom bypass edges provide additional short routes that avoid the damaged slice altogether. Each patch edge joins vertices at ambient Hamming distance two, so the repairs remain local in the coordinate geometry. Theorem 5.15 shows that these repairs restore the required connectivity while keeping the maximum degree of order $n^{1/k}$.

Theorem 5.15. *For every $k \geq 3$ and $n \geq k^k$, every graph $\widetilde{H} \in \mathcal{H}_{\mathrm{nb}}(n, k)$ is k-step k-connected and satisfies*

$$\Delta(\widetilde{H}) \leq (k+1)\lfloor n^{1/k} \rfloor.$$

Consequently, $\Delta^\star(n; k, k) = \Theta(n^{1/k})$.

Theorems 5.7 and 5.15, together with the lower bound in (5.3), determine the correct order of growth of the primary objective in (5.1). Since k-step k-connectivity implies k-step m-connectivity for every $m \leq k$, we obtain the following corollary.

Corollary 5.16. *Fix $k \geq 3$ and $2 \leq m \leq k$. Then $\Delta^\star(n; k, m) = \Theta(n^{1/k})$. More precisely, if $n \geq k^k$, then*

$$(n-1)^{1/k} \leq \Delta^\star(n; k, m) \leq (k+1)\lfloor n^{1/k} \rfloor.$$

Having determined the optimal maximum-degree scale, we next examine the edge economy of the same constructions. The following subsection shows that, after allowing one additional step, they satisfy a substantially stronger connectivity requirement while using asymptotically the minimum possible number of edges for that stronger requirement. These combined properties lead to the *Hydra family*.

5.2 Formal Properties of the Hydra Family

For fixed k, the previous subsection shows that the family $\mathcal{H}_{\mathrm{nb}}(n, k)$ achieves the optimal order of growth for the primary objective in (5.1): its maximum degree is of order $n^{1/k}$ while it remains k-step k-connected. We now show that the same construction has additional structure that becomes visible once one allows one extra step. This leads naturally to the following terminology.

Definition 5.17. For integers $k \geq 3$ and $n \geq k^k$, define

$$\mathrm{Hydra}(n, k) := \mathcal{H}_{\mathrm{nb}}(n, k).$$

Any graph $\widehat{H} \in \mathrm{Hydra}(n, k)$ is called a *Hydra graph*.

The name *Hydra* stands for *Hyperconnected, Decentralized, Rewiring Architecture*. The next theorem justifies the first two features.

Theorem 5.18. *Let $k \geq 3$, $n \geq k^k$, and let $\widehat{H} \in \mathrm{Hydra}(n,k)$. Let $q_1, \ldots, q_k$ be as balanced as possible for (n,k), and define $M_H := \sum_{\ell=1}^{k}(q_\ell - 1)$, the degree of the ambient Hamming graph $H(q_1, \ldots, q_k)$. Then:*

(i) *$\widehat{H}$ is k-step k-connected;*

(ii) *$\widehat{H}$ is $(k+1)$-step $(M_H - 1)$-connected;*

(iii) *$\delta(\widehat{H}) \geq M_H - 1 \geq k(\lfloor n^{1/k} \rfloor - 1) - 1$ and $\Delta(\widehat{H}) \leq (k+1)\lfloor n^{1/k} \rfloor$.*

Hyperconnectivity refers to the large increase in the number of guaranteed short independent paths after one additional step: the number of paths rises from the constant level k to the degree scale $n^{1/k}$. Decentralization refers to the fact that the maximum degree remains of optimal order, while the minimum degree is of the same order, so the construction avoids both hubs and extreme degree disparity.

This stronger connectivity also reveals the edge economy of the Hydra construction. For every $2 \leq m \leq k$, Hydra achieves the optimal order of the primary maximum-degree objective. More strikingly, the next theorem shows that its edge count is asymptotically minimal among all graphs satisfying the much stronger parameter pair $(k+1, M_H - 1)$, even without imposing a maximum-degree restriction. For integers $\ell, m \geq 2$, write

$$e^\star(n; \ell, m) := \min\bigl\{\|G\| : |G| = n,\ G \text{ is } \ell\text{-step } m\text{-connected}\bigr\}.$$

Theorem 5.19. *Assume $k \geq 3$ and $n \geq k^k$, and let $\widehat{H} \in \mathrm{Hydra}(n,k)$. Let $q_1, \ldots, q_k$ be as balanced as possible for (n,k), and set $M_H := \sum_{\ell=1}^{k}(q_\ell - 1)$. Then*

$$\frac{n(M_H - 1)}{2} \leq e^\star(n; k+1, M_H - 1) \leq \|\widehat{H}\| \leq \frac{n(M_H + k)}{2}. \tag{5.5}$$

Comparing the outer bounds in (5.5) yields

$$1 \leq \frac{\|\widehat{H}\|}{e^\star(n; k+1, M_H - 1)} \leq 1 + \frac{k+1}{M_H - 1} \longrightarrow 1 \qquad \text{as } n \to \infty,$$

since $M_H = \Theta(n^{1/k})$. Thus, after allowing one additional step, Hydra is asymptotically edge-optimal for the much stronger connectivity requirement $M_H - 1$.

The final feature behind the Hydra name is *rewiring*. Explicit update rules keep the graph within the Hydra family after a single-vertex deletion or insertion. Away from boundary cases, the changes involve only the affected vertices and the patch edges associated with one top-slice position; at a boundary, the ambient Hamming grid must instead be resized and reindexed.

For deletion, any vacancy outside the surviving top slice is transferred to it by relocating a surviving top-slice vertex, after which the usual patching mechanism is applied. For insertion, a vacant top-slice position is filled; if none exists, the graph is embedded in the next near-balanced Hamming grid. The precise update rules are given in Algorithms 3 and 4.

Algorithm 3 Hydra deletion repair

Require: $\widehat{H} \in \mathrm{Hydra}(n,k)$ with $n > k^k$, embedded in $H := H(q_1, \dots, q_k)$, where $q_1 \le \cdots \le q_k$; current deletion set D; deleted vertex v

Ensure: A graph in $\mathrm{Hydra}(n-1,k)$

Let $S := \{x \in V(H) : x_k = q_k\}$ be the ambient top slice and $T := S \setminus D$ the surviving reference slice immediately before the deletion.
if $v \notin T$ **then**
    Choose $w \in T$ and relocate the surviving vertex w to v.
    Remove every edge formerly incident to w.
    Restore every ambient Hamming edge from the relocated vertex at v to an occupied ambient neighbour.
    Restore every anchor-repair or top-to-bottom bypass edge formerly incident to v whose other endpoint remains occupied.
    Set $v \leftarrow w$.
end if
    The vacancy is now at a vertex v of the original surviving reference slice T.
if $T \setminus \{v\} \neq \emptyset$ **then**
    Set $D' := D \cup \{v\}$.
    Add the anchor-repair edges associated with the deletion of v.
    Replace D by D'.
else
    Set $(q'_1, \dots, q'_k) := (q_k - 1, q_1, \dots, q_{k-1})$. By Proposition 5.13, this tuple is as balanced as possible for $(n-1,k)$. Define $H' := H(q'_1, \dots, q'_k)$.
    Under the mapping

$$(x_1, \dots, x_k) \longmapsto (x_k, x_1, \dots, x_{k-1}),$$

    identify the induced subgraph $H \setminus S$ with H'.
    Let $S' := \{x \in V(H') : x_k = q'_k\}$ be the new ambient top slice, and set $D' := \emptyset$.
    Retain every ambient Hamming edge of H' and add the top-to-bottom bypass edges incident to the vertices of S'.
    Replace D by D'.
end if

Algorithm 4 Hydra insertion update

Require: $\widehat{H} \in \mathrm{Hydra}(n,k)$ embedded in $H := H(q_1,\dots,q_k)$, where $q_1 \le \cdots \le q_k$; current deletion set D; one new vertex

Ensure: A graph in $\mathrm{Hydra}(n+1,k)$

Let $S := \{x \in V(H) : x_k = q_k\}$ be the ambient top slice.
if $D \neq \emptyset$ **then**
    Choose $v \in D$ and place the new vertex at v.
    Set $D' := D \setminus \{v\}$.
    Remove the anchor-repair edges associated with the former deletion of v.
    Restore every ambient Hamming edge from v to an occupied ambient neighbour.
    For every $w \in D'$ adjacent to v in S, add the anchor-repair edge from v associated with the deletion of w.
    Add the top-to-bottom bypass edges incident to v.
    Replace D by D'.
else
    Set $(q'_1,\dots,q'_k) := (q_2,\dots,q_k,q_1+1)$. By Proposition 5.13, this tuple is as balanced as possible for $(n+1,k)$. Define $H' := H(q'_1,\dots,q'_k)$.
    Let $S' := \{x \in V(H') : x_k = q'_k\}$ be the new ambient top slice. Under the mapping

$$(x_1,\dots,x_k) \longmapsto (x_2,\dots,x_k,x_1),$$

    identify the old ambient Hamming graph with the induced subgraph $H' \setminus S'$.
    Choose $v \in S'$, place the new vertex at v, and set $D' := S' \setminus \{v\}$.
    Remove the old top-to-bottom bypass edges.
    Retain every ambient Hamming edge in $H' \setminus S'$ and add every ambient Hamming edge from v to $H' \setminus S'$.
    For every $w \in D'$ adjacent to v in S', add the anchor-repair edge from v associated with the deletion of w.
    Add the top-to-bottom bypass edges incident to v.
    Replace D by D'.
end if

Theorem 5.20. *The Hydra family is closed under the single-vertex deletion and insertion updates in Algorithms 3 and 4. More precisely, if* $\widehat{H} \in \mathrm{Hydra}(n,k)$*, then:*

(i) *if* $n > k^k$*, deleting one vertex and applying Algorithm 3 produces a graph in* $\mathrm{Hydra}(n-1,k)$*;*

(ii) *inserting one vertex and applying Algorithm 4 produces a graph in* $\mathrm{Hydra}(n+1,k)$*.*

Thus, following each update, Theorem 5.18 applies with the updated parameters, as does Theorem 5.19 whenever its hypotheses are satisfied. This regenerative closure is the final reason for the name *Hydra*: like the Lernaean Hydra, the architecture restores its defining structure after local damage.

Remark 5.21. Hydra naturally invites comparison with expander families, which are the standard hub-free graph families exhibiting strong global connectivity. Classical explicit constructions include the Margulis-Gabber-Galil graphs and Ramanujan families; see Hoory et al. (2006).

The two constructions address different design regimes. Classical expanders keep the degree bounded as the number of vertices grows and consequently have diameter of order $\log n$. Hydra instead fixes the admissible path length at k or $k+1$; by the Moore bound, this forces the degree to grow on the scale $n^{1/k}$. Its purpose is to guarantee nearly degree-many internally vertex-disjoint paths within this fixed hop budget, while also admitting explicit deletion and insertion updates. Spectral expansion alone does not provide either this precise finite-hop guarantee or the corresponding regenerative closure. We discuss the comparison further in Appendix D.

6 Applications and Implications

The Hydra family contains architectures with different maintenance, routing, and reliability properties, depending on the connectivity parameter k. The appropriate choice of k therefore depends on the cost structure of the application. We first study maintenance and reliability separately, then examine their interaction with usage costs under power-law activity, and finally apply the resulting framework to the Lightning Network.

6.1 Maintenance Costs and Minimizing Links

Suppose first that costs arise only from maintaining links. An edge is interpreted as a standing connectivity commitment, and there is no additional penalty for longer routes. The corresponding optimization problem is

$$\min\Big\{\|\widehat{H}\| : \widehat{H} \in \mathrm{Hydra}(n,k),\ k \geq 3,\ n \geq k^k\Big\}.$$

By Theorem 5.19, if $q_1, \ldots, q_k$ are as balanced as possible for (n,k) and $M_H := \sum_{\ell=1}^{k}(q_\ell - 1)$, then $n(M_H - 1) \leq 2\|\widehat{H}\| \leq n(M_H + k)$. By Proposition 5.13, $q_\ell \in \{\lfloor n^{1/k} \rfloor, \lfloor n^{1/k} \rfloor + 1\}$, and hence $M_H = kn^{1/k} + O(k)$. It follows that the number of links satisfies

$$\|\widehat{H}\| = \frac{n}{2} k n^{1/k} + O(nk).$$

The leading maintenance scale is therefore governed by $g_n(k) := kn^{1/k}$. Temporarily treating k as continuous gives

$$g_n'(k) = n^{1/k}\left(1 - \frac{\log n}{k}\right).$$

Every feasible value satisfies $k \geq 3$ and $k^k \leq n$, which implies $\log n \geq k \log k > k$. Consequently, $g_n'(k) < 0$ throughout the feasible range. The leading edge-count proxy is therefore minimized by the largest feasible dimension

$$K_n := \max\{k \in \mathbb{N} : k \geq 3,\ k^k \leq n\}.$$

The actual Hydra edge count also contains lower-order construction terms, so the preceding monotonicity does not by itself establish the same conclusion for every finite n. These terms are, however, asymptotically smaller than the maintenance-cost gap between K_n and the lower feasible dimensions. The next proposition therefore shows that K_n is eventually the exact edge-minimizing dimension.

Proposition 6.1. *For all sufficiently large n, the unique edge-minimizing dimension within the Hydra family is K_n. Moreover, let $\widehat{H}_n \in \mathrm{Hydra}(n, K_n)$ and let $M_n := \sum_{\ell=1}^{K_n}(q_\ell - 1)$, where $q_1, \ldots, q_{K_n}$ are as balanced as possible for (n, K_n). Then, as $n \to \infty$,*

$$K_n \sim \frac{\log n}{\log\log n}, \qquad M_n = K_n^2 + O(K_n \log K_n),$$

and

$$\|\widehat{H}_n\| = \left(\frac{1}{2} + o(1)\right) n K_n^2 = \left(\frac{1}{2} + o(1)\right) n \left(\frac{\log n}{\log\log n}\right)^2.$$

Proposition 6.1 shows that, for all sufficiently large networks, a designer concerned only with maintaining links chooses the largest feasible dimension K_n. The resulting maintenance cost remains near-linear, $\|\widehat{H}\| = n^{1+o(1)}$. This places Hydra between two familiar architectures: it is more expensive than centralized hub-based connectivity, which requires only $O(n)$ links, but vastly cheaper than full pairwise connectivity, which requires $O(n^2)$ links.

p	$k=3$	$k=4$	$k=5$	$k=6$	$k=7$
1/8	3.60%	2.93%	**2.74**%	2.81%	3.05%
1/3	**34.85**%	41.47%	49.36%	57.62%	65.56%

Table 3: The interaction-failure bound $\pi_k(p)$. Increasing k improves the bound when $p = 1/8$ up to a point, but worsens it when $p = 1/3$. The effect of increasing k is therefore ambiguous.

6.2 Reliability under Independent Edge Failures

Hydra does not rely on a distinguished hub. Its connectivity properties apply uniformly to every pair of vertices. Moreover, these properties are formulated in terms of internally vertex-disjoint paths and therefore already provide resilience to vertex deletions. They also yield explicit reliability bounds under independent edge failures.

Suppose that each edge fails independently with probability $p \in (0, 1)$. Fix two vertices x and y, and let $F_{x,y}$ denote the event that they can no longer interact after the edge failures. If $P_1, \ldots, P_r$ are edge-disjoint paths between x and y, with respective lengths $\ell_1, \ldots, \ell_r \le L$, then

$$\mathbb{P}[F_{x,y}] \le \mathbb{P}[P_1, \ldots, P_r \text{ all fail}] = \prod_{j=1}^{r} \left[1 - (1-p)^{\ell_j}\right] \le \left[1 - (1-p)^{L}\right]^{r}.$$

The equality follows because the paths are edge-disjoint and edge failures are independent. The first inequality may be strict because additional paths may remain operational.

We first use only the fact that Hydra is k-step k-connected. Every pair of vertices is joined by k edge-disjoint paths of length at most k, and hence

$$\mathbb{P}[F_{x,y}] \le \pi_k(p) := \left[1 - (1-p)^{k}\right]^{k}.$$

Increasing k has two opposing effects: it provides more alternative paths, but also permits each path to contain more edges. This dependence can be seen explicitly by temporarily treating k as continuous. The unique continuous minimizer of $\pi_k(p)$ is

$$k_p^{\text{cont}} = \frac{\log 2}{-\log(1-p)},$$

which is decreasing in p. For the admissible integer choices $k \ge 3$, the threshold at which $k = 3$ becomes optimal is determined by $\pi_3(p) = \pi_4(p)$. The unique solution is $p^* \approx 0.181$. Hence $k = 3$ minimizes the bound when $p \ge p^*$, whereas the minimizing value is strictly larger than 3 when $p < p^*$. Table 3 illustrates both regimes. When $p = 1/8$, the bound is minimized at $k = 5$. When $p = 1/3$, it is minimized at the smallest admissible value $k = 3$.

This calculation, however, discards most of the redundancy supplied by Hydra. Hyperconnectivity shows that allowing paths to use just one additional step produces at least $M_H - 1$ edge-disjoint paths between every pair of vertices; see Theorem 5.18. To that end, define

$$m_{n,k} := k(\lfloor n^{1/k} \rfloor - 1) - 1,$$

and note that $M_H - 1 \ge m_{n,k}$. Hyperconnectivity then gives

$$\mathbb{P}[F_{x,y}] \le \left[1 - (1-p)^{k+1}\right]^{M_H - 1} \le \bar{\pi}_{n,k}(p) := \left[1 - (1-p)^{k+1}\right]^{m_{n,k}}.$$

The second inequality gives a bound depending only on n, k, and p; using the exact value of M_H may produce an even stronger guarantee.

Unlike the basic bound, the two effects now point in the same direction. Lower-dimensional Hydra networks provide both shorter routes and greater hyperconnectivity-based redundancy. The next result formalizes this observation.

Panel A: $p = 1/8$						
n	K_n	$k = 3$	$k = 4$	$k = 5$	$k = 6$	$k = 7$
1,000	4	$\mathbf{1.09 \times 10^{-10}}$	2.06×10^{-5}	–	–	–
10,000	5	$\mathbf{2.47 \times 10^{-23}}$	1.17×10^{-11}	6.19×10^{-7}	–	–
100,000	6	$\mathbf{4.49 \times 10^{-52}}$	2.09×10^{-20}	4.15×10^{-12}	5.23×10^{-7}	–
1,000,000	7	$\mathbf{3.76 \times 10^{-114}}$	6.69×10^{-38}	1.41×10^{-18}	3.32×10^{-12}	3.19×10^{-8}
Panel B: $p = 1/3$						
n	K_n	$k = 3$	$k = 4$	$k = 5$	$k = 6$	$k = 7$
1,000	4	$\mathbf{3.27 \times 10^{-3}}$	0.12	–	–	–
10,000	5	$\mathbf{2.30 \times 10^{-6}}$	7.14×10^{-3}	0.11	–	–
100,000	6	$\mathbf{1.56 \times 10^{-13}}$	1.37×10^{-4}	1.75×10^{-2}	0.17	–
1,000,000	7	$\mathbf{5.14 \times 10^{-29}}$	5.04×10^{-8}	1.76×10^{-3}	4.09×10^{-2}	0.20

Table 4: The hyperconnectivity-based interaction-failure bound $\bar{\pi}_{n,k}(p)$. Bold entries indicate the minimum over the feasible displayed values of k. A dash indicates that $k^k > n$.

Proposition 6.2. *Fix $p \in (0,1)$. The hyperconnectivity-based bound satisfies the following monotonicity properties.*

(i) *For every fixed feasible dimension k, $\bar{\pi}_{n,k}(p)$ is nonincreasing in n.*

(ii) *For every fixed n, $\bar{\pi}_{n,k}(p)$ is strictly increasing in k over the range of feasible dimensions $3 \le k \le K_n$. Consequently, $\min_{3 \le k \le K_n} \bar{\pi}_{n,k}(p) = \bar{\pi}_{n,3}(p)$.*

Table 4 reports the bounds for the two illustrative edge-failure probabilities $p = 1/8$ and $p = 1/3$. Entries are omitted whenever $k^k > n$, since the corresponding Hydra architecture is infeasible. The comparison with Table 3 demonstrates the value of the additional step in the hyperconnectivity guarantee. For example, when $n = 1{,}000$, $k = 3$, and $p = 1/3$, using only 3-step 3-connectivity gives the bound

$$\pi_3(1/3) = 0.3485.$$

Accounting for paths of length four reveals at least 26 independent routes and reduces the corresponding bound to

$$\bar{\pi}_{1000,3}(1/3) = 3.274 \times 10^{-3}.$$

The reliability implication is therefore the opposite of the maintenance-cost implication. Once the full hyperconnectivity of Hydra is used, the strongest reliability guarantee is obtained from the smallest admissible value $k = 3$. By contrast, Proposition 6.1 shows that, for all sufficiently large networks, the Hydra edge count is uniquely minimized by the largest feasible value K_n. This tension motivates a joint cost model that accounts for link maintenance, hop-dependent usage, and losses from failed interactions under power-law network activity.

Remark 6.3. The magnitudes of the hyperconnectivity-based bounds are striking. When $n = 1{,}000$, $k = 3$, and $p = 1/3$, the interaction-failure probability is at most 3.27×10^{-3} for every fixed pair of vertices, even though each edge fails with probability $1/3$.

The independence assumption is also more natural for Hydra than for a hub-based architecture. Since Hydra has no distinguished vertex whose removal would cause widespread disruption, there is no obvious target around which failures should be concentrated. A star, by contrast, has a single point of failure. Removing its hub disconnects every pair of surviving leaves. Even under independent edge failures, two leaves in a star fail to interact with probability $1-(1-p)^2$, which equals $5/9$ when $p = 1/3$.

Finally, the underlying connectivity results are stronger than the edge-failure calculation suggests. The paths supplied by Hydra are internally vertex-disjoint, so they also provide resilience to vertex deletions. We focus on edge failures here because temporary link outages are the more common source of disruption in many network applications.

6.3 Maintenance, Usage, and Reliability under Power-Law Activity

Fix a common planning horizon and let T_n denote the total number of interactions occurring during that horizon. We suppose that aggregate activity follows the power law

$$T_n = \lambda n^{\alpha},$$

where $\lambda > 0$ measures traffic intensity and $\alpha \in [1, 2]$ determines how activity scales with network size. Because each interaction involves two vertices, the mean number of interactions in which a vertex participates is $2T_n/n = 2\lambda n^{\alpha-1}$.

The exponent α is a property of the interaction environment rather than a parameter chosen by the network designer. The linear regime $\alpha = 1$ is the natural choice when vertices represent individual human users whose activity is constrained by time, attention, or income. Superlinear activity may arise when interactions are automated or when the number of counterparties used by each participant increases with network size. At the extreme $\alpha = 2$, each vertex interacts with an order-n collection of other vertices over the planning horizon.

We now combine maintenance, usage, and losses from failed interactions in a single objective. Normalize the cost of maintaining one link over the planning horizon to one. Let $u > 0$ denote the cost of one hop relative to maintaining one link, and let $v > 0$ denote the loss generated by a failed interaction. If $N_{n,k}^{\mathrm{fail}}$ is the number of failed interactions, the hyperconnectivity-based interaction-failure bound gives

$$\mathbb{E}[N_{n,k}^{\mathrm{fail}}] \le T_n \bar{\pi}_{n,k}(p),$$

provided that the interacting pairs are determined independently of the edge failures. Using k as a proxy for the mean number of hops and replacing the number of maintained links by the Hydra edge scale gives the joint cost function

$$C_\alpha(k; n, \lambda, u, v, p) := \frac{n}{2} k n^{1/k} + uk\lambda n^{\alpha} + v\lambda n^{\alpha} \bar{\pi}_{n,k}(p).$$

The next theorem characterizes the asymptotically optimal choice of the Hydra dimension.

Theorem 6.4. *Fix $\lambda, u, v > 0$, $p \in (0, 1)$, and $\alpha \in [1, 2]$. Recall that*

$$K_n := \max\{k \in \mathbb{N} : k \ge 3,\ k^k \le n\} \sim \frac{\log n}{\log\log n}$$

is the largest feasible Hydra dimension. Then every sequence of minimizers

$$k_n^\star \in \operatorname*{arg\,min}_{k \in \{3, \dots, K_n\}} C_\alpha(k; n, \lambda, u, v, p)$$

is eventually uniquely determined as follows.

(i) *If $\alpha > 4/3$, then $k_n^\star = 3$ for all sufficiently large n.*

(ii) *If $1 < \alpha < 4/3$ and $1/(\alpha - 1) \notin \mathbb{N}$, then $k_n^\star = \lceil 1/(\alpha - 1) \rceil$ for all sufficiently large n.*

(iii) *If $\alpha = 1 + 1/j$ for some integer $j \ge 3$, then for all sufficiently large n,*

$$k_n^\star = \begin{cases} j + 1, & \lambda u < j/2, \\ j, & \lambda u \ge j/2. \end{cases}$$

(iv) *If $\alpha = 1$, then $k_n^\star = K_n$ for all sufficiently large n.*

The theorem identifies three economically distinct activity regimes. When $\alpha > 4/3$, usage dominates maintenance even at the smallest admissible dimension, so the shortest-route architecture $k = 3$ is eventually optimal. When $1 < \alpha < 4/3$, maintenance excludes dimensions for which $1+1/k > \alpha$, while usage selects the smallest remaining dimension. Away from the critical exponents, this gives $k_n^\star = \lceil 1/(\alpha - 1) \rceil$. At the boundary values $\alpha = 1 + 1/j$, the comparison between j and $j + 1$ is resolved by comparing λu with $j/2$. Finally, when $\alpha = 1$, maintenance dominates the linear usage cost and the largest feasible dimension K_n is eventually optimal.

Thus the asymptotic phase diagram is governed by the competition between maintenance and usage rather than reliability. For every fixed dimension, hyperconnectivity makes the failure-loss term vanish faster than any power of n; in the linear-activity regime, where the optimal dimension grows with n, this term is at most of order n and is dominated by the maintenance savings. Reliability may nevertheless affect the preferred architecture at finite network sizes. This finite-size consideration is central to the Lightning Network application.

6.4 Implications for the Lightning Network

The Lightning Network is a payment-channel network built on top of the Bitcoin blockchain. Two participants can open a bilateral channel through an on-chain transaction and subsequently exchange payments off-chain. Opening and maintaining a channel is nevertheless costly: it requires an on-chain transaction, committed bitcoin liquidity, and monitoring; cf. Poon and Dryja (2016); Guasoni et al. (2024); Guasoni and Khan (2024). Ordinary users therefore have an incentive to maintain only a small number of channels. Payments between participants without a direct channel must be routed through intermediate nodes. Each additional hop may incur a routing fee and introduces another channel whose liquidity may be insufficient for the payment; see Zabka et al. (2022); Saraswathi and Kümmerle (2025). The Lightning Network therefore exhibits the maintenance, route-length, and interaction-failure trade-offs studied above.

Decentralization is central to both Bitcoin and the Lightning Network, yet the latter's topology is generated by local channel-opening decisions. Locally attractive peer-selection rules can produce undesirable global outcomes; see, for example, Lange et al. (2021); Mahdizadeh et al. (2023). In particular, degree- and betweenness-based attachment favour nodes that are already central. Empirical studies accordingly find highly skewed degree distributions, with a few routing hubs and a large low-degree periphery; see Seres et al. (2020); Martinazzi and Flori (2020); Lin et al. (2020); Guasoni et al. (2025). This concentration places routing power, information and failure risk in the hands of a few nodes, whose targeted removal can sharply reduce network efficiency. A payment layer built to scale a decentralized currency thereby risks recreating the centralized points of control and failure that decentralization is intended to avoid.

Applying Hydra to the entire public Lightning Network is not feasible under current channel budgets. At the time of writing, the live statistics reported in Amboss Technologies (2026) give approximately 15,000 public nodes and 41,000 public channels. The corresponding mean public degree is approximately

$$\frac{2 \times 41{,}000}{15{,}000} \approx 5.47,$$

and the degree distribution is strongly right-skewed. For $n = 15{,}000$, the feasible Hydra dimensions are $k = 3, 4, 5$, with balanced coordinate sizes

$$(24, 25, 25), \qquad (11, 11, 11, 12), \qquad (7, 7, 7, 7, 7).$$

Theorem 5.18 therefore gives respectively 70, 40, and 29 edge-disjoint routes between every pair, which implies that even the least demanding architecture would require every participant to maintain at least 29 channels. This is incompatible with the liquidity and operational constraints faced by most ordinary users.

A more plausible design is a *Hydra routing layer* comprising professional routing and liquidity providers. The ingredients already exist in the form of Lightning Service Providers, described in Lightning Labs (2022) as entities that provide liquidity services on behalf of other Lightning users. Let V be the set of all n nodes and let $\mathcal{R} \subset V$, with $|\mathcal{R}| = N$, be the routing layer, whose induced graph satisfies

$$\widehat{H}_{N,k} \in \mathrm{Hydra}(N, k).$$

For simplicity, suppose that every access node $x \in V \setminus \mathcal{R}$ opens channels to d distinct routing nodes; heterogeneous access degrees can be incorporated without changing the argument. Distributing these attachments as evenly as possible across $\mathcal{R}$ prevents the access structure from creating new hubs. In the absence of direct channels between access nodes, the total number of channels is

$$|E| = \|\widehat{H}_{N,k}\| + d(n - N).$$

Ordinary users thus maintain only their chosen d channels, while the denser channel structure is maintained by providers equipped and compensated to supply routing and liquidity.

The preceding reliability bounds extend directly to this architecture. Suppose that every channel is independently unavailable with probability p, and let $F_{x,y}$ denote the failure of an interaction between two access nodes. Each endpoint retains at least one live access channel with probability $1 - p^d$. Conditional on both events, choose surviving routing nodes a and b attached to x and y, respectively. Using k-step k-connectivity within the routing layer gives

$$\mathbb{P}[F_{x,y}] \le 1 - (1 - p^d)^2\big(1 - \pi_k(p)\big), \qquad \pi_k(p) = \Big[1 - (1 - p)^k\Big]^k,$$

with route length at most $k + 2$. Using hyperconnectivity instead gives

$$\mathbb{P}[F_{x,y}] \le 1 - (1 - p^d)^2\big(1 - \bar{\pi}_{N,k}(p)\big),$$

where

$$\bar{\pi}_{N,k}(p) = \Big[1 - (1 - p)^{k+1}\Big]^{m_{N,k}}, \qquad m_{N,k} = k\big(\lfloor N^{1/k} \rfloor - 1\big) - 1,$$

and the route length is at most $k + 3$. The d access channels allow for entry and exit from the routing layer; they do not multiply the internal path count, since the path families associated with different access-node pairs may overlap.

For illustration, fix $n = 15{,}000$ and $p = 1/3$, and begin with a routing layer of size $N = 750$, which is 5% of the entire network. For $k = 3$, the balanced ambient dimensions are $(9, 9, 10)$, the number of internal channels lies between 9,000 and 10,500, and $m_{750,3} = 23$. For $k = 4$, the corresponding quantities are $(5, 5, 5, 6)$, between 6,000 and 7,875 internal channels, and $m_{750,4} = 15$. Table 5 compares the resulting channel and reliability bounds.

To examine the effect of enlarging the routing layer, suppose instead that $N = 1{,}500$, so that 10% of nodes are routing providers. For $k = 3$, the balanced ambient dimensions are $(11, 12, 12)$, the number of internal channels lies between 23,250 and 26,250, and $m_{1500,3} = 29$. For $k = 4$, the corresponding quantities are $(6, 6, 6, 7)$, between 15,000 and 18,750 internal channels, and $m_{1500,4} = 19$. Table 6 reports the resulting bounds.

The two tables expose three design margins. Increasing d improves access reliability but requires ordinary users to divide liquidity across more channels. For fixed N and d, $k = 4$ requires fewer channels, whereas $k = 3$ gives shorter routes and substantially stronger reliability bounds. Increasing N raises the number of channels maintained within the professional routing layer and spreads access demand across more providers, but also strengthens the hyperconnectivity guarantee. It does not affect the basic bound, which uses only k routes and is independent of N. The reliability gain from increasing N is modest when d is small because interaction failure is then driven mainly by access to the routing layer. The choices of N, k, and d should therefore

Panel A: $k = 3$			
Access degree d	Total channels	Basic bound	Hyperconnectivity bound
1	23,250–24,750	71.04%	55.84%
2	37,500–39,000	48.52%	21.49%
3	51,750–53,250	39.58%	7.86%
4	66,000–67,500	36.45%	3.07%
Panel B: $k = 4$			
Access degree d	Total channels	Basic bound	Hyperconnectivity bound
1	20,250–22,125	73.99%	60.90%
2	34,500–36,375	53.75%	30.49%
3	48,750–50,625	45.72%	18.42%
4	63,000–64,875	42.90%	14.19%

Table 5: Illustrative cost–reliability trade-offs for a Hydra routing layer with $n = 15{,}000$, $N = 750$, and $p = 1/3$. Access channels are allocated as evenly as possible among routing providers.

be made jointly, according to users' channel budgets and the number of providers capable of meeting the required capital and service standards.

Finally, the construction remains dynamic. The entry or departure of an access node changes only its own d channels, while the Hydra insertion and deletion rules determine how the routing layer should be rewired when a professional provider enters or leaves. More fundamentally, intermediation is distributed across a large, balanced routing layer rather than concentrated at a few dominant hubs. For fixed k, its guaranteed redundancy increases with N, and no routing provider occupies a structurally privileged position. Although the probabilistic bounds above concern independent channel failures, Hydra's vertex-disjoint routes also limit the effect of targeted node removals: disabling a few providers cannot sever the surviving routing layer. The guarantees remain topological and do not by themselves ensure sufficient directional liquidity for a particular payment, but they supply the global architectural objective missing from purely local attachment strategies – a redundant and dynamically maintainable routing layer whose resilience strengthens as the network grows.

7 Conclusion

This paper shows that short, failure-resilient connectivity need not come at the price of either hub dependence or a dense network. We formalize this requirement through k-step m-connectivity and, within the feasible class, minimize maximum degree and then edge count. This lexicographic objective first limits the reliance placed on any one vertex and then minimizes the total number of links.

The Moore bound shows that any such architecture must have maximum degree at least $(n-1)^{1/k}$; see Equation (5.3). Thus, when the hop budget is fixed, universal connectivity cannot be achieved while keeping every vertex degree bounded independently of the size of the network. Hydra shows that this unavoidable degree scale is nevertheless compatible with the other design requirements. For every $k \geq 3$ and $n \geq k^k$, Algorithm 1 gives an explicit architecture obtained by pruning a near-balanced Hamming graph and repairing the routes destroyed by the deleted vertices.

In the dimension-matched regime $m = k$, Theorem 5.18 shows that every pair of vertices in

Panel A: $k = 3$			
Access degree d	Total channels	Basic bound	Hyperconnectivity bound
1	36,750–39,750	71.04%	55.63%
2	50,250–53,250	48.52%	21.12%
3	63,750–66,750	39.58%	7.43%
4	77,250–80,250	36.45%	2.62%
Panel B: $k = 4$			
Access degree d	Total channels	Basic bound	Hyperconnectivity bound
1	28,500–32,250	73.99%	58.59%
2	42,000–45,750	53.75%	26.39%
3	55,500–59,250	45.72%	13.61%
4	69,000–72,750	42.90%	9.12%

Table 6: Illustrative cost–reliability trade-offs for a Hydra routing layer with $n = 15{,}000$, $N = 1{,}500$, and $p = 1/3$. Access channels are allocated as evenly as possible among routing providers.

Hydra is joined by k independent paths of length at most k, and that its degrees satisfy

$$M_H - 1 \le \delta(\widehat{H}) \le \Delta(\widehat{H}) \le (k+1)\lfloor n^{1/k} \rfloor,$$

where $M_H \ge k(\lfloor n^{1/k} \rfloor - 1)$ is the degree of the ambient Hamming graph. Hence, for fixed k, Hydra matches the maximum-degree order imposed by the Moore bound. Moreover, its degrees are comparable across vertices, so connectivity is not concentrated in a small number of hubs. Theorem 5.18(ii) further shows that allowing one additional hop increases the guaranteed number of independent paths from k to $M_H - 1$. Any graph satisfying this stronger hyperconnectivity has at least $n(M_H - 1)/2$ edges, whereas Theorem 5.19 gives $\|\widehat{H}\| \le n(M_H + k)/2$. Hydra's edge count is therefore asymptotically minimal for this requirement as $n \to \infty$.

These are not merely static properties. Hydra admits explicit procedures for deleting and inserting vertices. Theorem 5.20 shows that, whenever the updated network size remains at least k^k, these procedures return the graph to the appropriate Hydra family and recover its short-path resilience, degree bounds, hyperconnectivity, and asymptotic edge optimality. Hydra therefore provides both a target architecture and a method for maintaining it as participation changes.

The free parameter k introduces the central economic design choice in the framework. Maintenance and reliability point in opposite directions. If links have a common maintenance cost and there is no separate cost associated with route length or failed interactions, Proposition 6.1 shows that the Hydra edge count is eventually minimized by the largest feasible dimension

$$K_n := \max\{k \in \mathbb{N} : k \ge 3,\ k^k \le n\}.$$

In this case,

$$K_n \sim \frac{\log n}{\log\log n}, \qquad \|\widehat{H}_n\| = \left(\frac{1}{2} + o(1)\right) n K_n^2 = \Theta\left(n\left(\frac{\log n}{\log\log n}\right)^2\right).$$

Maintenance alone therefore favours the largest feasible value of k and produces a near-linear edge count.

Reliability considerations lead to a different conclusion. Under k-step k-connectivity, increasing k supplies more independent routes but also permits them to be longer, so the resulting interaction-failure bound may increase or decrease with k. Once Hydra's full hyperconnectivity

is used, however, lower dimensions provide both shorter routes and greater guaranteed redundancy. Proposition 6.2 shows that the corresponding bound is nonincreasing in n and strictly increasing in k, so this criterion favours $k = 3$. Tables 3 and 4 demonstrate the quantitative importance of hyperconnectivity.

The joint model of Section 6.3 combines link maintenance, hop-dependent usage, and losses from failed interactions under aggregate activity $T_n = \lambda n^{\alpha}$. Theorem 6.4 identifies three asymptotic regimes. When $\alpha = 1$, bounded activity leaves maintenance dominant and K_n is eventually optimal. When $1 < \alpha < 4/3$, the optimal dimension is fixed and intermediate, with $k_n^{\star} = \lceil 1/(\alpha - 1) \rceil$ away from the critical exponents. At the critical values $\alpha = 1 + 1/j$, the relative maintenance and usage cost determines whether j or $j + 1$ is selected. When $\alpha > 4/3$, usage dominates and $k = 3$ is eventually optimal. The failure-loss term does not affect this asymptotic phase diagram because hyperconnectivity makes it lower order, but it can affect architectural choices at finite network sizes.

Section 6.4 makes these finite-size trade-offs concrete. Applying Hydra to the entire public Lightning Network is incompatible with the liquidity and operational constraints faced by ordinary users. We therefore propose a Hydra routing layer of dimension k maintained by N professional routing and liquidity providers, with ordinary users opening only d access channels. The choices of N, k, and d then expose distinct cost–reliability margins. For example, Table 5 shows that, with $n = 15{,}000$, $N = 750$, $k = 3$, and $p = 1/3$, taking $d = 2$ gives between 37,500 and 39,000 total channels and a hyperconnectivity-based interaction-failure bound of 21.49%. Increasing the access degree to $d = 3$ raises the channel count to between 51,750 and 53,250, but reduces the bound to 7.86%. Comparing Tables 5 and 6 shows how enlarging the routing layer strengthens internal redundancy, although the improvement is modest when access failure is the dominant source of disruption. The Hydra update procedures also specify how the routing layer should be rewired as providers enter and leave.

The paper is a first step toward an economic theory of decentralized connectivity, but it goes beyond identifying a topological feasibility frontier. It shows how explicit maintenance, usage, and reliability criteria select among Hydra architectures and how the same principles can inform a concrete routing-layer design. Further work can incorporate directed and state-dependent liquidity, correlated failures, heterogeneous capacities and fees, and endogenous traffic. The central economic question is whether decentralized incentives can sustain such a global architecture without recreating the dependence on dominant hubs that it is intended to avoid.

A Definitions, Notation and Terminology

This appendix recalls the basic concepts and notation used in the paper, following the standard conventions of Diestel (2025).

A *graph* is an ordered pair $G = (V, E)$, where $V \neq \emptyset$ and $E \subset \{\{x, y\} : x, y \in V,\ x \neq y\}$. The elements of V and E are called the *vertices* and *edges* of G, respectively. The number of vertices is the *order* of G, denoted by $|G|$, while the number of edges is its *size*, denoted by $\|G\|$.

- Two vertices $x, y \in V$ are *adjacent* if $\{x, y\} \in E$; an edge is said to be *incident* to its two endpoints. A graph in which every pair of distinct vertices is adjacent is the *complete graph* on n vertices, denoted K_n.
- The *degree* of a vertex v, denoted $\deg_G(v)$ or simply $\deg(v)$, is the number of edges incident to v. A graph is called *regular* if all vertices have the same degree.
- The *minimum* and *maximum degrees* are

$$\delta(G) \coloneqq \min_{v \in V} \deg(v) \quad \text{and} \quad \Delta(G) \coloneqq \max_{v \in V} \deg(v).$$

  The *average degree* is

$$\overline{d}(G) \coloneqq \frac{1}{|V|} \sum_{v \in V} \deg(v).$$

  By the handshaking lemma,

$$2|E| = \sum_{v \in V} \deg(v) = \overline{d}(G)|V|.$$

- The *neighborhood* of a vertex v is

$$N_G(v) := \{u \in V : \{u, v\} \in E\}.$$

- A *walk* of length $t \geq 1$ from $x \in V$ to $y \in V$ is a sequence $(v_0, \ldots, v_t)$ such that $v_0 = x$, $v_t = y$, and $\{v_{\ell-1}, v_\ell\} \in E$ for every $\ell \in \{1, \ldots, t\}$. An x–y *path* is a walk from x to y whose vertices are all distinct. Two x–y paths are *independent*, or *internally vertex-disjoint*, if they share no vertices other than x and y. A collection of paths is independent if its members are pairwise independent.
- The *distance* between two vertices $x, y \in V$, denoted $\operatorname{dist}_G(x, y)$, is the length of the shortest path joining them (with $\operatorname{dist}_G(x, y) = \infty$ if no such path exists). A path between x and y of length $\operatorname{dist}_G(x, y)$ is called a *geodesic* between x and y. The *diameter* of G is

$$\operatorname{diam}(G) \coloneqq \max_{x, y \in V} \operatorname{dist}_G(x, y).$$

- For $U \subset V$, let $G[U]$ denote the subgraph of G induced by U.
- Let $G = (V_G, E_G)$ and $H = (V_H, E_H)$ be two graphs. Their *Cartesian product* $G \,\square\, H$ is the graph with vertex set $V(G \,\square\, H) = V_G \times V_H$ and edge set

$$E(G \,\square\, H) = \Big\{\{(g, h), (g', h')\} : \big(g = g' \text{ and } \{h, h'\} \in E_H\big) \text{ or } \big(h = h' \text{ and } \{g, g'\} \in E_G\big)\Big\}.$$

B k-Step m-Connectedness

B.1 Related Notions

A graph $G = (V, E)$ is *connected* if every two vertices are joined by a path; otherwise it is disconnected. The *connectivity* $\kappa(G)$ is the minimum number of vertices whose removal disconnects G or leaves a trivial graph, and G is *m-connected* if $\kappa(G) \geq m$. Equivalently, an m-connected graph remains connected after the removal of any $m-1$ vertices. A classical theorem of Menger (1927) gives a path-based characterization of this notion:

Theorem (Menger's Theorem). *A graph is m-connected if and only if every pair of vertices is joined by m independent paths.*

In many applications, however, this notion is too weak because it does not control path length. Even if an m-connected graph initially has small diameter, deleting a few vertices can dramatically increase distances. Indeed, (Chung and Garey, 1984, Theorem 7) shows that, in the worst case, removing $s < m$ vertices from an m-connected graph of order n can increase the diameter by a factor of

$$\frac{\lfloor (n-s-2)/(m-s) \rfloor + 1}{2}.$$

For example, if $n = 1000$, $m = 5$, and $s = 1$, then the diameter may increase by a factor of 125. This shows that classical m-connectivity does not by itself control post-failure distances. Once path length matters, two natural bounded-length analogues of Menger's theorem emerge:

(i) after removing any $m-1$ vertices, every remaining pair of vertices is still connected by a path of length at most k;

(ii) every pair of vertices is connected by at least m independent paths of length at most k.

Both notions reduce to classical m-connectivity when $k = \infty$. Moreover, (ii) implies (i), since removing $m-1$ vertices cannot destroy all m independent short paths. The converse, however, fails in general. The subtle relationship between local versions of these two bounded-length notions is studied in Lovász et al. (1978); Pyber and Tuza (1993).

There is also an extremal literature for property (i). Given natural numbers n, d, d', s with $s < n$ and $d \leq d' \leq n-1$, let $\mathcal{G}(n, d, d', s)$ denote the set of graphs of order n and diameter at most d such that removing any s vertices leaves a graph of diameter at most d'. Define

$$f(n, d, d', s) \coloneqq \min\{\|G\| : G \in \mathcal{G}(n, d, d', s)\}.$$

This line of work was initiated by Murty and Vijayan (1964); see also (Bollobás, 2004, Chapter 4) and Caccetta (1979). A graph satisfies property (i) if and only if it belongs to $\mathcal{G}(n, k, k, m-1)$, so this corresponds to the extremal problem $f(n, k, k, m-1)$. The cases $k = 2$ and $m \geq 2$, and $k \in \{3, 4\}$ with $m = 2$, are relatively well understood (Bollobás, 2004, Section 4.3), but beyond these regimes much less is known.

A weaker edge analogue is obtained by replacing vertex deletions with edge deletions:

(i)' after removing any $m-1$ edges, every remaining pair of vertices is still connected by a path of length at most k.

Slightly more is known in this setting. In particular, the cases $k = 2$ and $m \geq 2$ (Bollobás, 2004, Section 4.3), and $k \geq 3$ with $m = 2$ Chung (1992), are well understood.

In this paper we work with the stronger condition (ii), which we call *k-step m-connectivity*. This notion is closely related to the *wide diameter*, introduced independently in Flandrin and Li (1994); Hsu (1994). For an m-connected graph G, the m-wide diameter $d_m(G)$ is the smallest integer ℓ such that every pair of vertices is joined by m independent paths of length at most ℓ. Thus G is k-step m-connected if and only if $d_m(G) \leq k$. For a broader overview of wide diameter and its computation in standard graph families, see Xu (2001).

B.2 Related Extremal Problems

To understand the role of our objective (5.1), it is helpful to compare it with several classical related extremal problems.

(1) Classical connectivity: no length bound. When $k = \infty$, the constraint of being k-step m-connected reduces to ordinary m-connectivity by Menger's theorem. The classical extremal problem is

$$\min\{\|G\| : |G| = n,\ G \text{ is } m\text{-connected}\}.$$

For $m = 1$, the minimum is $n - 1$, attained by every tree. For $2 \le m \le n - 1$, Harary (1962) showed that the minimum is $\lceil mn/2 \rceil$, attained by the Harary graphs $H_{m,n}$; see Figure 3(b). Thus, in the unbounded-length setting, the edge-minimal problem is completely understood.

Moreover, when $k = \infty$, our problem

$$\min\{(\Delta(G), \|G\|) : |G| = n,\ G \text{ is } \infty\text{-step } m\text{-connected}\}$$

is essentially aligned with this classical edge-minimization problem. For $m \ge 2$, Harary graphs minimize the number of edges and also achieve the smallest possible maximum degree, namely m when mn is even and $m + 1$ when mn is odd. The only exception is $m = 1$: edge-minimality allows any tree, whereas the optimum is attained by a path. Thus the tension between sparsity and decentralization does not yet arise when $k = \infty$.

(2) A natural extension: bounded diameter under m-connectivity. A natural strengthening of classical m-connectivity is to impose a diameter bound:

$$g(n, d, m) \coloneqq \min\{\|G\| : |G| = n,\ G \text{ is } m\text{-connected and } \operatorname{diam}(G) \le d\}, \qquad d \ge 2.$$

For $m = 1$, the minimum is still $n - 1$, attained by any tree of diameter at most d; a star is one such example; see Figure 3(a). For $m = 2$ and n sufficiently large, the problem is solved in Belotserkovsky (2007). For $m \ge 3$, asymptotically optimal constructions were developed by Bollobás and Harary (1976) and are discussed further in (Bollobás, 2004, Chapter 4). These constructions can be viewed as sparse Harary-type cores guaranteeing connectivity, augmented by a small set of *connectors* that enforce the diameter bound.

For example, when $d = 2$, one may take an $(m-1)$-connected Harary graph on $n-1$ vertices and add a single vertex adjacent to all others; see Figure 3(c). The resulting graph has diameter 2 and the number of edges

$$E(n) = \left\lceil \frac{(n-1)(m-1)}{2} \right\rceil + (n-1).$$

In particular,

$$\frac{2E(n)}{n} \longrightarrow m + 1 \qquad \text{as } n \to \infty,$$

so the average degree remains bounded. Nevertheless, the added connector has degree $n - 1$. For $d > 2$, the single hub is replaced by a carefully designed connector set, but the same phenomenon persists. Bounded diameter is enforced by vertices whose degrees grow linearly with n. Thus edge-efficient bounded-diameter connectivity is already naturally associated with centralization.

(3) Exact edge minimization in the two-step regime. For $k = 2$, the relevant extremal problem becomes

$$\min\{\|G\| : |G| = n,\ G \text{ is 2-step } m\text{-connected}\}.$$

Here the picture is even sharper. Murty (1968) proved that if $n \ge m(3 + \sqrt{5})/2$, then

$$\min\{\|G\| : |G| = n,\ G \text{ is 2-step } m\text{-connected}\} = m(n - m) + \binom{m}{2}.$$

Moreover, equality is attained uniquely by the graph obtained by partitioning $V = H \cup (V \setminus H)$ with $|H| = m$ and joining every pair of vertices except those both lying in $V \setminus H$; see Figure 3(d). In this construction every vertex in H has degree $n - 1$. Thus, in the two-step regime, exact edge-minimality is achieved by an explicitly hub-dominated architecture. The complementary range $n < m(3 + \sqrt{5})/2$ is less relevant for our applications, which lie in the large-scale regime $n \gg m$; for that range, see Bollobás and Erdös (1975) and (Faudree et al., 2012, Section 4).

(4) General k-step m-connectivity. For $k \geq 3$, exact edge-minimal values are known only in limited regimes. A basic benchmark is

$$\left\lceil \frac{mn}{2} \right\rceil \leq \min\{\|G\| : |G| = n,\ G \text{ is } k\text{-step } m\text{-connected}\} \leq m(n - m), \tag{B.1}$$

valid for $n \geq 2m$.

The lower bound follows because k-step m-connectivity forces the minimum degree $\delta(G) \geq m$, and hence $\|G\| \geq mn/2$ by the handshake lemma. The upper bound is obtained by taking a set H of m vertices and joining every vertex outside H to all vertices in H, with no further edges; see Figure 3(e). This graph is 3-step m-connected, hence k-step m-connected for every $k \geq 3$, and its vertices in H have degree $n - m$. So the generic constructive upper bound is again achieved by a centralized design.

On the other hand, (Faudree et al., 2012, Section 5) identifies several parameter ranges with $k \geq 3$ and $m \geq 2$ in which the lower bound in (B.1) is attained. In these cases there exist k-step m-connected graphs with exactly $\lceil mn/2 \rceil$ edges, and the extremal examples may be chosen m-regular. Thus edge-minimality need not always force hubs. However, these positive results occur when n is of the same order as m, often under additional parity or divisibility restrictions. They do not cover the large-scale regime $n \gg m$ that is most relevant for our applications. Indeed, (Faudree et al., 2012, Corollary 6.3) shows that the lower bound in (B.1) cannot remain sharp once n is sufficiently large relative to m.

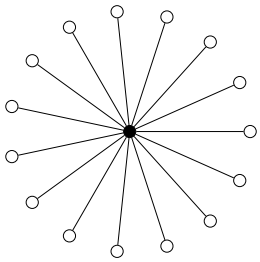
(5) Why decentralization must come first. Taken together, these benchmark problems reveal a common pattern. When n is large and path length is constrained, edge-efficient constructions typically rely on a small set of vertices with degree growing linearly in n. In other words, sparsity alone tends to produce hubs.

This is undesirable in many applications. Hub-based networks are well known to be structurally fragile because removing a few high-degree vertices can drastically impair connectivity Albert et al. (2000); Motter and Lai (2002). They also concentrate traffic and betweenness on a small subset of vertices, which increases vulnerability to congestion and overload cascades Goh et al. (2001). More generally, in growing networks preferential attachment generates hubs and then reinforces their dominance over time, leading to persistent degree inequality Barabási and Albert (1999). These considerations motivate our choice of objective. If one wants a sparse architecture without such concentration, decentralization must be treated as the primary design criterion rather than as a by-product of edge minimization.

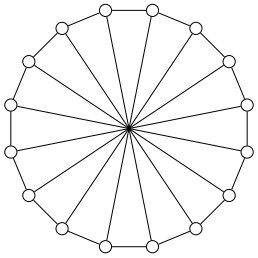
C An Integer Linear Programming Formulation

At a purely conceptual level, (5.1) could be solved by exhaustive search. There are $2^{\binom{|V|}{2}}$ labelled simple graphs on V. With unlimited computational power, one could enumerate them all, test whether each graph is k-step m-connected, and return the graph(s) with lexicographically smallest pair $(\Delta(G), \|G\|)$. The integer linear program (ILP) below expresses this finite search as a single optimization problem.

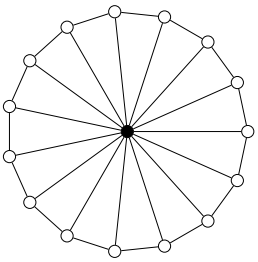
The nontrivial part is encoding k-step m-connectedness. For each pair of distinct vertices $s, t \in V$, this property requires the existence of m simultaneously chosen, internally vertex-disjoint s–t paths, each of length at most k. The auxiliary variables introduced below encode this existential requirement as a system of linear constraints.



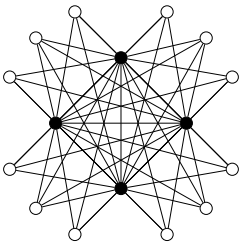
(a) Star ($m = 1$).



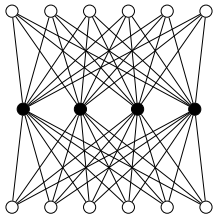
(b) Harary $H_{3,16}$ (3-regular).



(c) $H_{2,15}$ with a central hub.



(d) m-hub clique star ($m = 4$).



(e) m-hub independent star ($m = 4$).

Figure 3: Edge-efficient designs often rely on hubs (black nodes), except in the unbounded-diameter Harary case (b). Constructions scale to general n but are shown for $n = 16$ for clarity.

Fix integers $n \geq 2$ and $k, m \geq 1$, and let

$$V := \{1, \dots, n\}, \qquad \mathcal{E} := \binom{V}{2}, \qquad P := \{(s,t) \in V \times V : s < t\}.$$

Thus, P contains exactly one ordered representative of each unordered pair in $\mathcal{E}$. We also set

$$R := \{1, \dots, m\}, \qquad L := \{1, \dots, k\}, \qquad L_0 := \{0, \dots, k\}.$$

For notational convenience, whenever $u \neq v$, set $x_{uv} := x_{\min\{u,v\},\max\{u,v\}}$. The following mixed-integer linear program gives an exact formulation of (5.1). The design variables are

$$\begin{aligned} &x_{ij} \in \{0,1\} && \{i,j\} \in \mathcal{E}, && \big(\tbinom{n}{2} \text{ variables}\big),\\ &d_i \in \{0, \dots, n-1\} && i \in V, && \big(n \text{ variables}\big),\\ &\Delta \in \{0, \dots, n-1\}, && && \big(1 \text{ variable}\big). \end{aligned}$$

Here, x_{ij} indicates whether the edge $\{i,j\}$ is present, d_i records the degree of vertex i, and Δ is an upper bound on the maximum degree. Thus, the graph selected by x is

$$G_x := (V, E_x), \qquad E_x := \big\{\{i,j\} \in \mathcal{E} : x_{ij} = 1\big\},$$

with

$$\|G_x\| = \sum_{\{i,j\} \in \mathcal{E}} x_{ij}.$$

For every $(s,t) \in P$ and $r \in R$, we introduce an expanded representation of the r-th s–t route:

$$\begin{aligned} &a_v^{st,r,\ell} \in \{0,1\} && \ell \in L_0,\ v \in V, && \big(\tbinom{n}{2} m(k+1)n \text{ variables}\big),\\ &y_{uv}^{st,r,\ell} \in \{0,1\} && \ell \in L,\ u,v \in V,\ u \neq v, && \big(\tbinom{n}{2} mkn(n-1) \text{ variables}\big). \end{aligned}$$

The variable $a_v^{st,r,\ell}$ indicates that route r is at vertex v after ℓ steps, while $y_{uv}^{st,r,\ell}$ indicates that its ℓ-th step moves from u to v. Although the installed edges are undirected, the step variables are directed.

Let $M := \binom{n}{2} + 1$. The objective is

$$\min \quad M\Delta + \sum_{\{i,j\}\in\mathcal{E}} x_{ij}. \tag{C.1}$$

The degree variables satisfy

$$d_i = \sum_{j\in V\setminus\{i\}} x_{ij}, \qquad i \in V, \tag{C.2}$$

$$d_i \le \Delta, \qquad i \in V. \tag{C.3}$$

Each indexed route is initialized at s:

$$a_s^{st,r,0} = 1, \qquad (s,t)\in P,\ r\in R, \tag{C.4}$$

$$a_v^{st,r,0} = 0, \qquad (s,t)\in P,\ r\in R,\ v\in V\setminus\{s\}. \tag{C.5}$$

Its movement through successive layers is governed by

$$a_v^{st,r,\ell} = \sum_{u\in V\setminus\{v\}} y_{uv}^{st,r,\ell}, \qquad (s,t)\in P,\ r\in R,\ \ell\in L,\ v\in V, \tag{C.6}$$

$$\sum_{v\in V\setminus\{u\}} y_{uv}^{st,r,\ell} = a_u^{st,r,\ell-1}, \qquad (s,t)\in P,\ r\in R,\ \ell\in L,\ u\in V\setminus\{t\}, \tag{C.7}$$

$$\sum_{v\in V\setminus\{t\}} y_{tv}^{st,r,\ell} = 0, \qquad (s,t)\in P,\ r\in R,\ \ell\in L. \tag{C.8}$$

Finally, the following constraints enforce arrival, edge activation, and disjointness:

$$\sum_{\ell=1}^{k} a_t^{st,r,\ell} = 1, \qquad (s,t)\in P,\ r\in R, \tag{C.9}$$

$$\sum_{\ell=1}^{k} a_s^{st,r,\ell} = 0, \qquad (s,t)\in P,\ r\in R, \tag{C.10}$$

$$y_{uv}^{st,r,\ell} \le x_{uv}, \qquad (s,t)\in P,\ r\in R,\ \ell\in L,\ u,v\in V,\ u\ne v, \tag{C.11}$$

$$\sum_{r\in R}\sum_{\ell=1}^{k-1} a_v^{st,r,\ell} \le 1, \qquad (s,t)\in P,\ v\in V\setminus\{s,t\}, \tag{C.12}$$

$$\sum_{r\in R} y_{st}^{st,r,1} \le x_{st}, \qquad (s,t)\in P. \tag{C.13}$$

The objective (C.1) is an exact scalarization of the lexicographic objective in (5.1). Indeed,

$$M = \binom{n}{2} + 1 > \binom{n}{2} \ge \|G_x\|,$$

so any reduction in Δ dominates every possible change in $\|G_x\|$. Among solutions with the smallest value of Δ, the objective therefore minimizes the number of edges. Constraints (C.2)–(C.3) define the vertex degrees and impose $\Delta \ge \Delta(G_x)$; equality holds at every optimal solution.

We next explain the routing constraints. Constraints (C.4)–(C.5) place every indexed route at s at layer 0. Constraint (C.7) forces the route to take exactly one outgoing step from its current vertex unless it has reached t, while (C.6) records its location at the next layer. Constraint (C.8) ensures that the route terminates upon reaching t, and (C.9) requires this to occur within k steps. Constraint (C.10) rules out a subsequent return to s, while (C.11) permits a step only along an installed edge.

Constraint (C.12) ensures that each internal vertex is visited at most once across all m indexed routes. It therefore both prevents a route from repeating an internal vertex and prevents two different routes from sharing one. The only possible duplication not detected by this constraint is the length-one path consisting of the edge $\{s, t\}$, since that path has no internal vertices. Constraint (C.13) ensures that this path is selected at most once.

It follows that every feasible solution defines, for each $(s, t) \in P$, m independent s–t paths of length at most k, all contained in G_x. Conversely, given m such paths for every pair, the corresponding a- and y-variables can be set equal to one along those paths and to zero elsewhere. The projection of the feasible region onto the x-variables is therefore exactly the set of k-step m-connected graphs on V.

The formulation is exact but large. Writing $p := \binom{n}{2}$, its total number of variables is

$$p + n + 1 + pm(k + 1)n + pmkn(n - 1).$$

In particular, it contains $pm(k+1)n = \Theta(mkn^3)$ location variables and $pmkn(n-1) = \Theta(mkn^4)$ step variables. Hence the total number of variables is $\Theta(mkn^4)$.

The number of scalar constraints is

$$2n + pmn + 2pmkn + 2pm + pmkn(n - 1) + p(n - 1) = \Theta(mkn^4).$$

The six terms correspond, respectively, to the degree constraints, route initialization, route movement, terminal conditions, edge activation, and the combination of internal-vertex disjointness and direct-path uniqueness. In particular, the edge-activation family (C.11) alone contains $pmkn(n - 1)$ inequalities. Thus, the size of the formulation is dominated by the directed step variables and their associated edge-activation constraints.

For example, if $n = 1000$, $m = 4$, and $k = 4$, then

$$|P| = \binom{1000}{2} = 499{,}500.$$

The formulation contains 9,990,000,000 binary location variables and 7,984,008,000,000 binary step variables. Including the design and degree variables, this gives 7,993,998,500,501 variables in total. The formulation also contains 8,002,492,998,500 scalar constraints, of which 7,984,008,000,000 are instances of (C.11). Direct instantiation is therefore far beyond the memory and branch-and-bound capacity of general-purpose ILP solvers. The formulation also exhibits substantial symmetry, both from permutations of the m path indices and from relabellings of isomorphic graph designs.

For small n, the formulation remains useful for validating and benchmarking candidate constructions. For larger instances, a decomposition or branch-and-cut approach would be required: one may retain the graph-design variables in a master problem, test candidate designs using bounded-length internally vertex-disjoint path subproblems, and add valid inequalities or lazy constraints when a connectivity requirement is violated. Such methods avoid instantiating all routing variables simultaneously, although the separation subproblems and the generation of strong cuts may themselves become computational bottlenecks.

These computational limitations, together with the limited structural information provided by numerical solutions at individual values of n, motivate the explicit constructions developed in this paper, which provide uniform analytical guarantees across different network sizes.

D Expander Graphs

A family (G_n) of finite d-regular graphs is called an expander family if $|G_n| \to \infty$ and there exists $h_0 > 0$ such that the Cheeger constant

$$h(G_n) := \min_{1 \le |S| \le |V(G_n)|/2} \frac{|\partial S|}{|S|}$$

satisfies $h(G_n) \geq h_0$ for all n. For regular graphs, this geometric definition is equivalent to a uniform spectral gap. Indeed, if $\lambda_2(G)$ denotes the second-largest adjacency eigenvalue of a d-regular graph G, then Hoory et al. (2006, Theorem 2.4) show that

$$\frac{d - \lambda_2(G)}{2} \leq h(G) \leq \sqrt{2d\big(d - \lambda_2(G)\big)}.$$

Thus, for fixed d, a family of d-regular graphs is an expander family if and only if $d - \lambda_2(G_n)$ is bounded away from zero uniformly in n.

This spectral point of view yields an immediate estimate for the number of walks. Let A be the adjacency matrix of a d-regular graph G with $N := |V(G)|$, and let

$$\lambda := \max_{i \geq 2} |\lambda_i(A)|.$$

Since $(A^\ell)_{uv}$ is the number of walks of length ℓ from u to v, spectral decomposition gives

$$\left|(A^\ell)_{uv} - \frac{d^\ell}{N}\right| \leq \lambda^\ell, \qquad u, v \in V(G),\ \ell \geq 0.$$

For the Hamming benchmark $H(q, \dots, q)$ of dimension k, one has $N = q^k$, $d_H = k(q-1)$, and adjacency eigenvalues

$$\vartheta_j = d_H - qj, \qquad j = 0, \dots, k;$$

see Brouwer and Haemers (2012, Section 12.3.1). Hence, for $q \geq 3$,

$$\left|(A_H^\ell)_{uv} - \frac{d_H^\ell}{q^k}\right| \leq \lambda_H^\ell, \qquad \lambda_H := \max_{1 \leq j \leq k} |\vartheta_j| = \max\{d_H - q, k\}.$$

(Indeed, one can say more for Hamming graphs, because their distance-regular structure yields exact formulas depending only on Hamming distance.)

At a superficial level, these estimates suggest that bounded-degree expanders may offer connectivity properties comparable to Hamming graphs while keeping degree bounded. But this analogy is misleading for our purposes. The bounds above count *walks*, not paths: walks may revisit vertices, and the estimates give no control over internal vertex-disjointness. Standard expander theory does imply small diameter and strong global connectivity, but it does not furnish the kind of pairwise lower bounds on short, internally vertex-disjoint paths that are required in our notion of decentralized connectivity.

This distinction is central here. Our design criterion is k-step m-connectivity, which is defined in terms of independent paths, not generic walks. In particular, Theorem 5.15 gives a precise guarantee of k independent paths of length at most k between every pair of vertices, while Theorem 5.18 shows that allowing one extra step raises the number of independent short paths to the degree scale $M_H - 1 = \Theta(n^{1/k})$. Moreover, the Hydra construction comes with an explicit delete-and-rewire repair rule: after node removals, the surviving vertices can replace lost links so as to recover the same decentralized connectivity guarantees. No analogous restoration mechanism is part of the definition of an expander family. For this reason, the relevant comparison is not that Hydra "behaves like" an expander in terms of walk counts, but that Hydra provides the stronger object needed in this paper: explicit lower bounds on the number of short independent paths, together with an explicit self-healing procedure.

E Supplementary Results and Proofs

Proof of Proposition 5.5. (1) Since $c = \lceil n/r \rceil \geq n/r$, multiplying by r gives $rc \geq n$.

(2) From $r = \lfloor \sqrt{n} \rfloor$ we have $r^2 \le n < (r+1)^2 = r^2 + 2r + 1$. Since n is an integer, $r^2 \le n \le r^2 + 2r$. Therefore $r \le n/r \le r + 2$, and since r is an integer,

$$c = \lceil n/r \rceil \in \{r, r+1, r+2\}.$$

(3) Let $s := \lceil \sqrt{n} \rceil$. If $n = s^2$ then $r = c = s$ and $r + c - 2 = 2s - 2$. Otherwise $r = s - 1$ and, by (2), $c \le r + 2 = s + 1$, so

$$r + c - 2 \le (s-1) + (s+1) - 2 = 2s - 2 = 2\lceil \sqrt{n} \rceil - 2.$$

(4) From $c = \lceil n/r \rceil$ we have $c - 1 < n/r \le c$. Multiplying by r yields $rc - r < n \le rc$, hence

$$0 \le t = rc - n < r. \qquad \square$$

Proof of Theorem 5.7. The Moore lower bound (5.3) gives $\Delta^\star(n; 2, 2) \ge (n-1)^{1/2}$, so it remains to show that every $G \in \mathcal{R}_{\mathrm{ns}}(n)$ is 2-step 2-connected and satisfies $\Delta(G) \le 2\lceil \sqrt{n} \rceil - 2$.

Set $r := \lfloor \sqrt{n} \rfloor$, $c := \left\lceil \frac{n}{r} \right\rceil$ and $t := rc - n$. By Proposition 5.5, $c \in \{r, r+1, r+2\}$, $t < r$ and $r + c - 2 \le 2\lceil \sqrt{n} \rceil - 2$. Write

$$d_\alpha := (\alpha, \alpha), \qquad \alpha = 1, \dots, t,$$

for the deleted vertices. We view G as obtained from the ambient rook graph $R_{r,c}$ by deleting $d_1, \dots, d_t$ and then adding the prescribed patch edges.

Degree bound. Every vertex of $R_{r,c}$ has degree $r + c - 2$. Since the deleted vertices lie in distinct rows and distinct columns, a surviving vertex $x = (i, j)$ loses at most one neighbour in its row and at most one neighbour in its column. If $j \le t$, then x loses the column edge to d_j and receives exactly one patch edge as a surviving vertex in column j. If $i \le t$, then x loses the row edge to d_i and receives at most one patch edge as a selected target in row i. Thus the number of added patch edges incident to x is never larger than the number of deleted ambient edges incident to x, and at least one patch edge is added whenever x loses two ambient edges. Consequently,

$$r + c - 3 \le \deg_G(x) \le r + c - 2 \qquad \text{for all } x \in V(G).$$

In particular,

$$\Delta(G) \le r + c - 2 \le 2\lceil \sqrt{n} \rceil - 2.$$

2-step 2-connectivity. Let $x \ne y$ be vertices of G.

If x and y lie in a common row or a common column, then they are adjacent. Since $r \ge 4$ and $c \ge r$, that row or column contains at least one further surviving vertex, so x and y also have a common neighbor. Hence they are joined by two independent paths of length at most 2.

Now suppose $x = (i, j)$ and $y = (i', j')$ lie in different rows and different columns. In $R_{r,c}$, their two crossing vertices

$$z_1 := (i, j'), \qquad z_2 := (i', j)$$

are common neighbors.

If neither z_1 nor z_2 is deleted, we are done.

If exactly one crossing is deleted, say

$$z_1 = (i, j') = d_\alpha = (\alpha, \alpha),$$

then $i = \alpha$ and $j' = \alpha$, so

$$y = (i', j') = (i', \alpha)$$

lies in the deleted column of d_α. By construction, the patch for d_α joins y to some surviving vertex w in row $\alpha = i$. If $w \ne x$, then w is a common neighbor of x and y; together with the

surviving crossing z_2, this gives two distinct common neighbors. If $w = x$, then the patch edge gives the direct path $x - y$, and the surviving crossing z_2 gives a second path $x - z_2 - y$. These two paths are independent and both have length at most 2.

Finally, suppose both crossings are deleted. Then

$$z_1 = d_\alpha, \qquad z_2 = d_\beta$$

for distinct α, β, so

$$x = (\alpha, \beta), \qquad y = (\beta, \alpha).$$

The patch for d_α joins y to some surviving vertex w_α in row α, and by the defining restriction on the patching we have $w_\alpha \neq x$. Thus w_α is a common neighbor of x and y. Similarly, the patch for d_β joins x to some surviving vertex w_β in row β, with $w_\beta \neq y$. Hence w_β is another common neighbor. Since w_α and w_β lie in different rows, they are distinct. Thus x and y again have two distinct common neighbors.

So every pair of distinct vertices in G is joined by two independent paths of length at most 2. Therefore G is 2-step 2-connected. □

Proof of Proposition 5.10. Fix two vertices $x = (x_1, \dots, x_k)$, $y = (y_1, \dots, y_k)$, and define

$$D := \{\ell \in \{1, \dots, k\} : x_\ell \neq y_\ell\}, \qquad E := \{\ell \in \{1, \dots, k\} : x_\ell = y_\ell\}.$$

Let $d := |D| \in \{1, \dots, k\}$. We will construct k independent $x - y$ paths of length at most k.

Geodesics along the differing coordinates. Write $D = \{\ell_1 < \cdots < \ell_d\}$. For $s = 1, \dots, d$, let

$$\rho^{(s)} = (\ell_s, \ell_{s+1}, \dots, \ell_d, \ell_1, \dots, \ell_{s-1})$$

and let $G^{(s)}$ be the path that changes the coordinates in D in the order $\rho^{(s)}$, replacing x_ℓ by y_ℓ at each step. Then $G^{(s)}$ is a geodesic from x to y of length d.

We claim that $G^{(1)}, \dots, G^{(d)}$ are independent. For $r \in \{1, \dots, d-1\}$, let $v^{(s)}(r)$ be the vertex reached after r steps on $G^{(s)}$, and define

$$F(v) := \{\ell \in D : \text{ the } \ell\text{-th coordinate of } v \text{ equals } y_\ell\}.$$

Then

$$F\big(v^{(s)}(r)\big) = \{\rho^{(s)}(1), \dots, \rho^{(s)}(r)\}.$$

Hence if $v^{(s)}(r) = v^{(s')}(r')$, then $|F(v^{(s)}(r))| = |F(v^{(s')}(r'))|$, so $r = r'$; and then equality of the sets forces $s = s'$, since $r < d$. Thus the internal vertices of the geodesics are all distinct.

Case 1: $d = k$. The k geodesics $G^{(1)}, \dots, G^{(k)}$ already give k independent $x - y$ paths, each of length k.

Case 2: $1 \leq d \leq k - 2$. Take the d geodesics $G^{(1)}, \dots, G^{(d)}$. Since $|E| = k - d$, we may choose $k - d$ distinct coordinates $t \in E$. For each such t, pick

$$c_t \in \{1, \dots, q_t\} \setminus \{x_t\},$$

which is possible because $q_t \geq 2$, and define a path $P^{(t)}$ by:

(i) change $t : x_t \to c_t$;

(ii) change each coordinate in D from x_ℓ to y_ℓ in any fixed order;

(iii) change $t : c_t \to x_t$.

Since $t \in E$, we have $y_t = x_t$, so $P^{(t)}$ is indeed an x–y path. Its length is

$$1 + d + 1 = d + 2 \leq k.$$

Every internal vertex of $P^{(t)}$ has t-th coordinate equal to $c_t \neq x_t$. By contrast, every internal vertex of a geodesic $G^{(s)}$ has t-th coordinate equal to x_t, because geodesics only modify coordinates in D. Likewise, if $t' \neq t$, then every internal vertex of $P^{(t')}$ also has t-th coordinate equal to x_t. Hence the paths $P^{(t)}$ are pairwise independent and are independent from all the geodesics. Altogether this gives

$$d + (k - d) = k$$

independent x–y paths of length at most k.

Case 3: $d = k - 1$. Take the $k - 1$ geodesics $G^{(1)}, \ldots, G^{(k-1)}$. For the final path, choose any $r \in D$ and pick

$$c_r \in \{1, \ldots, q_r\} \setminus \{x_r, y_r\},$$

which is possible because $q_r \geq 3$. Define a path Q by:

(i) change $r : x_r \to c_r$;

(ii) for each $\ell \in D \setminus \{r\}$, change $x_\ell \to y_\ell$ in any fixed order;

(iii) change $r : c_r \to y_r$.

This is an x–y path of length

$$1 + (k - 2) + 1 = k.$$

Every internal vertex of Q has r-th coordinate equal to c_r, whereas every internal vertex of a geodesic has r-th coordinate in $\{x_r, y_r\}$. Thus Q is independent from all $k - 1$ geodesics, giving k independent x–y paths of length at most k.

In all cases there are k independent x–y paths of length at most k. Therefore $H(q_1, \ldots, q_k)$ is k-step k-connected. □

Proof of Proposition 5.13. Since $r = \lfloor n^{1/k} \rfloor$, we have $r^k \leq n < (r+1)^k$. Hence there exists at least one $s \in \{0, 1, \ldots, k\}$ such that $Q_s = r^{k-s}(r+1)^s \geq n$. Let s be the smallest such integer, and define

$$q_1 = \cdots = q_{k-s} = r, \qquad q_{k-s+1} = \cdots = q_k = r + 1.$$

We first show that $(q_1, \ldots, q_k)$ is as balanced as possible for (n, k). Since $n \geq 2^k$, we have $r \geq 2$, so all the constructed side lengths are at least 2. Property (ii) in Definition 5.12 is immediate, since each q_ℓ is either r or $r + 1$. Property (i) also holds, because

$$\prod_{\ell=1}^{k} q_\ell = r^{k-s}(r+1)^s = Q_s \geq n.$$

It remains to prove the minimality in Definition 5.12(iii). Let $(p_1, \ldots, p_k)$ be any integer k-tuple with $p_\ell \geq 2$ and $\prod_{\ell=1}^{k} p_\ell \geq n$. For a fixed sum, the product is maximized when the entries differ by at most one. Indeed, whenever $p_i \leq p_j - 2$, replacing the pair (p_i, p_j) by $(p_i + 1, p_j - 1)$ preserves the sum and increases the product.

Suppose that $\sum_{\ell=1}^{k} p_\ell < kr + s$. If this sum is less than kr, then the arithmetic–geometric mean inequality gives

$$\prod_{\ell=1}^{k} p_\ell \leq \left(\frac{\sum_{\ell=1}^{k} p_\ell}{k} \right)^k < r^k \leq n,$$

a contradiction. Otherwise, $\sum_{\ell=1}^{k} p_\ell = kr + u$ for some $u \in \{0, \dots, s-1\}$. The largest product with this sum is attained by $k-u$ entries equal to r and u entries equal to $r+1$, and is therefore $r^{k-u}(r+1)^u = Q_u < n$, again contradicting admissibility. Hence $\sum_{\ell=1}^{k} p_\ell \geq kr + s = \sum_{\ell=1}^{k} q_\ell$. Thus $(q_1, \dots, q_k)$ satisfies Definition 5.12(iii).

Now let

$$Q := \prod_{\ell=1}^{k} q_\ell = Q_s, \qquad t := Q - n.$$

(1) We have already shown that $Q \geq n$, and by construction each $q_\ell \in \{r, r+1\}$.

(2) Since each $q_\ell \in \{r, r+1\}$, $r - 1 \leq q_\ell - 1 \leq r$ for all ℓ, and summing over $\ell = 1, \dots, k$ gives

$$k(r-1) \leq \sum_{\ell=1}^{k} (q_\ell - 1) \leq kr.$$

(3) If $s = 0$, then $Q = r^k$. Since also $r^k \leq n$ and $Q \geq n$, it follows that $Q = n$, so $t = 0$.

Suppose now that $s \geq 1$. By minimality of s, we have $Q_{s-1} < n \leq Q_s$. Therefore

$$0 \leq t = Q_s - n < Q_s - Q_{s-1}.$$

A direct computation gives

$$Q_s - Q_{s-1} = r^{k-s}(r+1)^s - r^{k-(s-1)}(r+1)^{s-1} = r^{k-s}(r+1)^{s-1},$$

and hence $0 \leq t < r^{k-s}(r+1)^{s-1}$. □

Proposition E.1. *Let $k \geq 3$ and $n \geq k^k$, and let $\widetilde{H} \in \mathcal{H}_{\mathrm{nb}}(n, k)$. Then*

$$\delta(\widetilde{H}) \geq M_H - 1 \geq k\big(\lfloor n^{1/k} \rfloor - 1\big) - 1, \qquad \Delta(\widetilde{H}) \leq (k+1)\lfloor n^{1/k} \rfloor.$$

Proof. Set $r := \lfloor n^{1/k} \rfloor$ and $H := H(q_1, \dots, q_k)$, where $q_1, \dots, q_k$ are as balanced as possible for (n, k). Let $S \subset V(H)$ be the top slice and $D \subset S$ the deletion set. By Proposition 5.13, $q_\ell \in \{r, r+1\}$. Write

$$M_H := \sum_{\ell=1}^{k} (q_\ell - 1),$$

so every vertex of H has degree M_H, and $k(r-1) \leq M_H \leq kr$.

Maximum degree. We distinguish four types of vertices.

Vertices outside the top and bottom slices that are not anchors. Such vertices are incident to no added edges. They lose at most one neighbor when D is deleted, and hence

$$\deg_{\widetilde{H}}(x) \leq M_H.$$

Surviving top-slice vertices. Every $x \in S \setminus D$ receives exactly $q_1 - 1$ bypass edges. For each deleted top-slice neighbor, it loses one ambient edge and gains exactly one anchor-repair edge. Consequently,

$$\deg_{\widetilde{H}}(x) = M_H + (q_1 - 1) \leq kr + r = (k+1)r.$$

Bottom-slice vertices. A bottom-slice vertex receives at most $q_1 - 1$ bypass edges. Therefore

$$\deg_{\widetilde{H}}(x) \leq M_H + (q_1 - 1) \leq (k+1)r.$$

Anchor vertices. An anchor $v^{(-a)}$ loses the ambient edge to v and gains $|N_a(v) \setminus D|$ anchor-repair edges. Since $|N_a(v)| = q_a - 1$, we have

$$\deg_{\widetilde{H}}\left(v^{(-a)}\right) \le M_H - 1 + (q_a - 1) \le kr + r = (k+1)r.$$

Taking the maximum over these four cases gives

$$\Delta(\widetilde{H}) \le (k+1)r = (k+1)\lfloor n^{1/k} \rfloor.$$

Minimum degree. Every surviving top-slice vertex has degree at least M_H, since each edge lost to a deleted top-slice neighbor is replaced by an anchor-repair edge. Every vertex outside S has at most one neighbor in D, namely the top-slice vertex with the same first $k-1$ coordinates. Since adding patch edges can only increase its degree, every such vertex has degree at least $M_H - 1$. Hence

$$\delta(\widetilde{H}) \ge M_H - 1 \ge k(r-1) - 1 = k\big(\lfloor n^{1/k} \rfloor - 1\big) - 1. \qquad \square$$

Proof of Theorem 5.15. By (5.3) and Proposition E.1, it remains to prove that every $\widetilde{H} \in \mathcal{H}_{\mathrm{nb}}(n,k)$ is k-step k-connected.

Fix $k \ge 3$ and $n \ge k^k$, and let $\widetilde{H} \in \mathcal{H}_{\mathrm{nb}}(n,k)$. Choose side lengths $q_1, \dots, q_k$ as balanced as possible for (n,k), and let

$$H := H(q_1, \dots, q_k), \qquad S := \{z \in V(H) : z_k = q_k\}.$$

Let $D \subset S$ be the deletion set. If $D = \emptyset$, then $\widetilde{H}$ contains H as a spanning subgraph, so the result follows from Proposition 5.10. We may therefore assume that $D \ne \emptyset$.

Set $r := \lfloor n^{1/k} \rfloor$. Since $n \ge k^k$, $r \ge k$. By Proposition 5.13, $q_\ell \ge r \ge k$ for $\ell < k$ and $q_k = r + 1 \ge k + 1$. We show that every two distinct vertices $x, y \in V(\widetilde{H})$ are joined by k independent paths of length at most k.

Case 1: $x, y \in V(\widetilde{H}) \setminus S$. The subgraph induced by $V(\widetilde{H}) \setminus S$ is $H(q_1, \dots, q_{k-1}, q_k - 1)$. All its side lengths are at least $k \ge 3$. Hence Proposition 5.10 gives k independent x–y paths of length at most k.

Case 2: $x, y \in S \setminus D$. For a horizontal vector $u = (u_1, \dots, u_{k-1})$ and a level $b \in \{1, \dots, q_k\}$, write $u[b] := (u_1, \dots, u_{k-1}, b)$. For $z \in V(H)$, write $\bar{z} := (z_1, \dots, z_{k-1})$ for its horizontal projection.

Regard the top slice as the horizontal Hamming graph $H(q_1, \dots, q_{k-1})$. The construction in the proof of Proposition 5.10 gives $k-1$ independent $\bar{x}$–$\bar{y}$ paths $R^{(1)}, \dots, R^{(k-1)}$ of length at most $k-1$. For each s, lifting every vertex u of $R^{(s)}$ to $u[q_k]$ gives an x–y path in the top slice. We now modify those lifted paths that contain a vertex of D.

Fix one such path and write

$$R^{(s)} = (u_0, u_1, \dots, u_L), \qquad u_0 = \bar{x}, \quad u_L = \bar{y}, \quad L \le k-1.$$

Let $j \in \{1, \dots, L-1\}$ be the smallest index such that $u_j[q_k] \in D$, and suppose that the step $u_{j-1} \to u_j$ changes horizontal axis a. Since $u_{j-1}[q_k]$ survives, the construction contains the anchor-repair edge

$$u_{j-1}[q_k] \to u_j[q_k - a].$$

After using this edge, follow the remaining horizontal changes prescribed by $R^{(s)}$ at level $q_k - a$, stopping at $u_{L-1}[q_k - a]$. Let b be the horizontal axis changed by the final step $u_{L-1} \to u_L$. If $u_{L-1}[q_k] \notin D$, complete the path with

$$u_{L-1}[q_k - a] \to u_{L-1}[q_k] \to u_L[q_k] = y.$$

If $u_{L-1}[q_k] \in D$, complete it instead with

$$u_{L-1}[q_k - a] \to u_{L-1}[q_k - b] \to u_L[q_k] = y.$$

Here $u_{L-1}[q_k - b]$ is the anchor associated with the deleted vertex $u_{L-1}[q_k]$ and axis b. Since the final step of $R^{(s)}$ changes axis b, the vertex $y = u_L[q_k]$ is a surviving top-slice neighbor of $u_{L-1}[q_k]$ along that axis. Therefore the construction includes the anchor-repair edge

$$u_{L-1}[q_k - b] \to y.$$

Thus, in the first case, the modified path is

$$u_0[q_k], \dots, u_{j-1}[q_k],\ u_j[q_k - a], \dots, u_{L-1}[q_k - a],\ u_{L-1}[q_k],\ u_L[q_k],$$

while in the second case $u_{L-1}[q_k]$ is replaced by $u_{L-1}[q_k - b]$. Consequently, the modification increases the length by at most one, and the resulting path has length at most $L + 1 \le k$.

We now verify independence. Every internal vertex z of the modified path satisfies

$$\bar{z} \in \{u_1, \dots, u_{L-1}\},$$

so its horizontal projection is an internal vertex of the original horizontal path $R^{(s)}$. In the first case, the two vertices $u_{L-1}[q_k - a]$ and $u_{L-1}[q_k]$ have the same horizontal projection u_{L-1}, but this causes no problem: u_{L-1} is an internal vertex belonging only to $R^{(s)}$. The same observation applies to $u_{L-1}[q_k - b]$ in the second case.

Since the original horizontal paths $R^{(1)}, \dots, R^{(k-1)}$ are independent, their sets of internal horizontal vertices are pairwise disjoint. Therefore, if two modified paths shared an internal vertex z, then its horizontal projection $\bar{z}$ would be an internal vertex of both corresponding horizontal paths, which is impossible. Hence the modified paths remain independent.

All their non-top-slice vertices have level $q_k - a$ or $q_k - b$ for some $a, b \in \{1, \dots, k-1\}$. These levels are at least 2, so none of the modified paths visits the bottom slice.

For the final path, choose $c \in \{1, \dots, q_1\} \setminus \{x_1, y_1\}$, which is possible because $q_1 \ge 3$. Use the bypass edge

$$x \to (c, x_2, \dots, x_{k-1}, 1),$$

change coordinates $2, \dots, k-1$ from their x-values to their y-values whenever necessary, and then use the bypass edge

$$(c, y_2, \dots, y_{k-1}, 1) \to y.$$

This path has length at most $1 + (k-2) + 1 = k$. All its internal vertices lie in the bottom slice, so it is independent of the preceding $k-1$ paths. Thus x and y are joined by k independent paths of length at most k.

Case 3: $x \in S \setminus D$ and $y \in V(\widetilde{H}) \setminus S$. Write $x = (x_1, \dots, x_{k-1}, q_k)$, $y = (y_1, \dots, y_k)$, and define

$$I := \{\ell \in \{1, \dots, k-1\} : x_\ell \ne y_\ell\}, \qquad J := \{\ell \in \{1, \dots, k-1\} : x_\ell = y_\ell\}.$$

Let $d := |I| \le k - 1$. We consider two subcases.

Case 3(a): $y_k = 1$. Choose k distinct values $c \in \{1, \dots, q_1\}$, which is possible since $q_1 \ge k$. For each such c, construct the path that begins with

$$x \to (c, x_2, \dots, x_{k-1}, 1).$$

If $c = x_1$, this is an ambient vertical edge; otherwise it is a bypass edge. Next change each coordinate $\ell \in I \setminus \{1\}$ from x_ℓ to y_ℓ in any order, and finally, if $c \ne y_1$, change the first coordinate from c to y_1. Each path has length at most $1 + (k-2) + 1 = k$. Every internal vertex on the path indexed by c has first coordinate c. Since the k chosen values of c are distinct, these paths are pairwise independent.

Case 3(b): $2 \le y_k \le q_k - 1$. We first construct $k-1$ paths that avoid the bottom slice. Write

$$I = \{\ell_1 < \cdots < \ell_d\}.$$

For each $s \in \{1, \dots, d\}$, construct a path $P^{(s)}$ as follows:

(i) Change coordinate ℓ_s from x_{ℓ_s} to y_{ℓ_s} using the corresponding anchor-repair edge if the intended top-slice vertex is deleted;

(ii) Change the k-th coordinate to y_k, omitting this step if the path is already at level y_k;

(iii) Change the remaining coordinates in I in the cyclic order $\ell_{s+1}, \dots, \ell_d, \ell_1, \dots, \ell_{s-1}$.

Each $P^{(s)}$ has length at most $d+1 \le k$.

If $d \le k-3$, then for each $t \in J$, choose $c_t \in \{1, \dots, q_t\} \setminus \{x_t\}$ and construct a path $P^{(t)}$ by:

(i) changing coordinate t from x_t to c_t, using the corresponding anchor-repair edge if the intended top-slice vertex is deleted;

(ii) changing the k-th coordinate to y_k, omitting this step if the path is already at level y_k;

(iii) changing every coordinate in I from its x-value to its y-value in any order;

(iv) changing coordinate t from c_t back to $x_t = y_t$.

These paths have length at most $d+3 \le k$. Together with the d paths indexed by I, they give $d + |J| = k-1$ paths.

If $d = k-2$, choose any $\ell^\star \in I$ and

$$c^\star \in \{1, \dots, q_{\ell^\star}\} \setminus \{x_{\ell^\star}, y_{\ell^\star}\}.$$

In addition to the d paths above, construct one path by changing coordinate $\ell^\star$ first from $x_{\ell^\star}$ to $c^\star$, changing the k-th coordinate to y_k, changing the coordinates in $I \setminus \{\ell^\star\}$, and finally changing coordinate $\ell^\star$ from $c^\star$ to $y_{\ell^\star}$. Use the corresponding anchor-repair edge if the first intended top-slice vertex is deleted and omit the second step if the path is already at level y_k. This path has length at most $1+1+(d-1)+1 = d+2 = k$. Thus there are again $d+1 = k-1$ paths.

Finally, if $d = k-1$, the paths $P^{(1)}, \dots, P^{(k-1)}$ already give the required $k-1$ paths.

These $k-1$ paths are independent. For the paths indexed by I, the coordinates already changed to their y-values form distinct cyclic initial segments of $(\ell_1, \dots, \ell_d)$, exactly as in the proof of Proposition 5.10. The paths indexed by J are distinguished by a coordinate t that retains the marker value c_t at every internal vertex. The additional path used when $d = k-2$ is similarly distinguished by the marker $c^\star$.

It remains to construct one path through the bottom slice. If $d \le k-2$, use the ambient vertical edge

$$x \to (x_1, \dots, x_{k-1}, 1),$$

change the coordinates in I to their y-values, and then move vertically to y. This path has length $d+2 \le k$.

If $d = k-1$, use the bypass edge

$$x \to (y_1, x_2, \dots, x_{k-1}, 1),$$

change coordinates $2, \dots, k-1$ to their y-values, and then move vertically to y. This path has length k.

In either case, all internal vertices of this final path lie in the bottom slice, whereas the preceding $k-1$ paths avoid the bottom slice. Hence all k paths are independent and have length at most k.

Thus every two distinct vertices of $\widetilde{H}$ are joined by k independent paths of length at most k. Therefore $\widetilde{H}$ is k-step k-connected. □

Proposition E.2. *Let $k \geq 2$ and $q_1, \ldots, q_k \geq 3$. Set $n := \prod_{\ell=1}^{k} q_\ell$ and $M_H := \sum_{\ell=1}^{k}(q_\ell - 1)$. Then the Hamming graph $H(q_1, \ldots, q_k)$ is $(k+1)$-step M_H-connected and solves*

$$\min\big\{\|G\| : |G| = n,\ G \text{ is } (k+1)\text{-step } M_H\text{-connected}\big\}.$$

Proof. Let $H := H(q_1, \ldots, q_k)$. Every vertex of H has degree M_H, so

$$\|H\| = \frac{nM_H}{2}.$$

Any $(k+1)$-step M_H-connected graph is in particular M_H-connected, hence has minimum degree at least M_H. Therefore every graph G with $|G| = n$ that is $(k+1)$-step M_H-connected satisfies

$$\|G\| \geq \frac{nM_H}{2} = \|H\|.$$

It remains to prove that H itself is $(k+1)$-step M_H-connected.

Fix two distinct vertices $x = (x_1, \ldots, x_k)$, $y = (y_1, \ldots, y_k)$, and define

$$D := \{\ell \in \{1, \ldots, k\} : x_\ell \neq y_\ell\}, \qquad E := \{\ell \in \{1, \ldots, k\} : x_\ell = y_\ell\}.$$

Let $d := |D| \leq k$. We construct M_H independent x–y paths of length at most $k+1$.

Geodesics along the differing coordinates. As in the proof of Proposition 5.10, there are d independent geodesics $G^{(1)}, \ldots, G^{(d)}$ from x to y, each of length $d \leq k$.

Marker paths on coordinates in D. Fix $\ell \in D$ and choose $c \in \{1, \ldots, q_\ell\} \setminus \{x_\ell, y_\ell\}$, which is possible since $q_\ell \geq 3$. Define a path $P^{(\ell,c)}$ by:

(i) change the ℓ-th coordinate from x_ℓ to c;

(ii) for each $m \in D \setminus \{\ell\}$, change x_m to y_m in any fixed order;

(iii) change the ℓ-th coordinate from c to y_ℓ.

This is an x–y path of length $1 + (d-1) + 1 = d + 1 \leq k + 1$. For fixed ℓ, there are $q_\ell - 2$ such choices of c.

Marker paths on coordinates in E. Fix $t \in E$ and choose $c \in \{1, \ldots, q_t\} \setminus \{x_t\}$, which is possible since $q_t \geq 2$. Define a path $Q^{(t,c)}$ by:

(i) change the t-th coordinate from x_t to c;

(ii) for each $m \in D$, change x_m to y_m in any fixed order;

(iii) change the t-th coordinate from c back to $x_t = y_t$.

This is an x–y path of length $1 + d + 1 = d + 2 \leq k + 1$, because $t \in E$ implies $d \leq k - 1$. For fixed t, there are $q_t - 1$ such choices of c.

Independence and count. The geodesics $G^{(s)}$ are pairwise independent by Proposition 5.10. Every internal vertex of $P^{(\ell,c)}$ has ℓ-th coordinate equal to $c \notin \{x_\ell, y_\ell\}$, so it cannot lie on a geodesic, on a path $P^{(\ell',c')}$ with $(\ell', c') \neq (\ell, c)$, or on a path $Q^{(t,c')}$. Likewise, every internal vertex of $Q^{(t,c)}$ has t-th coordinate equal to $c \neq x_t = y_t$, so it cannot lie on any other path. Hence all these paths are pairwise independent. Moreover, the total number of paths is

$$d + \sum_{\ell \in D}(q_\ell - 2) + \sum_{t \in E}(q_t - 1) = \sum_{\ell=1}^{k}(q_\ell - 1) = M_H.$$

Thus H is $(k+1)$-step M_H-connected. □

Proof of Theorem 5.18. Properties (i) and (iii) follow from Theorem 5.15 and Proposition E.1. It remains to prove (ii).

Let $q_1, \dots, q_k$ be the side lengths used in the construction of $\widehat{H}$, and set

$$H := H(q_1, \dots, q_k), \qquad S := \{z \in V(H) : z_k = q_k\}.$$

Let $D \subset S$ be the deletion set. If $D = \emptyset$, then $\widehat{H}$ contains H as a spanning subgraph. By Proposition E.2, H is $(k+1)$-step M_H-connected, so $\widehat{H}$ is $(k+1)$-step $(M_H - 1)$-connected.

We may therefore assume that $D \neq \emptyset$. Set $r := \lfloor n^{1/k} \rfloor$. Since $D \neq \emptyset$, we have $t > 0$, and Proposition 5.13 gives $q_k = r + 1$. Moreover, $q_\ell \geq r \geq k$ for $\ell < k$ and $q_k \geq k + 1$. Let

$$M_h := \sum_{\ell=1}^{k-1} (q_\ell - 1),$$

so that $M_H - 1 = M_h + (q_k - 2)$. For a horizontal vector $u = (u_1, \dots, u_{k-1})$ and a level $b \in \{1, \dots, q_k\}$, write $u[b] := (u_1, \dots, u_{k-1}, b)$. For a vertex $z \in V(H)$, write $\bar{z} := (z_1, \dots, z_{k-1})$ for its horizontal projection.

We show that every two distinct vertices $x, y \in V(\widehat{H})$ are joined by at least $M_H - 1$ independent paths of length at most $k + 1$. There are three cases.

Case 1: $x, y \in V(\widehat{H}) \setminus S$. The induced subgraph on $V(\widehat{H}) \setminus S$ is the Hamming graph

$$H(q_1, \dots, q_{k-1}, q_k - 1).$$

Its degree parameter is $\sum_{\ell=1}^{k-1}(q_\ell - 1) + ((q_k - 1) - 1) = M_H - 1$. Since $q_\ell \geq 3$ for $\ell < k$ and $q_k - 1 \geq 3$, Proposition E.2 shows that this subgraph is $(k+1)$-step $(M_H - 1)$-connected. Hence x and y are joined by $M_H - 1$ independent paths of length at most $k + 1$.

Case 2: $x, y \in S \setminus D$. We construct a horizontal family and a bottom-slice family.

Horizontal family. Apply the construction in the proof of Proposition E.2 to the horizontal Hamming graph

$$H(q_1, \dots, q_{k-1})$$

with endpoints $\bar{x}$ and $\bar{y}$. It gives M_h independent $\bar{x}$–$\bar{y}$ paths of length at most $(k-1)+1 = k$.

These paths may be indexed by

$$\mathcal{A} := \big\{(i, c) : i \in \{1, \dots, k-1\},\ c \in \{1, \dots, q_i\} \setminus \{y_i\}\big\}.$$

Indeed, $|\mathcal{A}| = \sum_{i=1}^{k-1}(q_i - 1) = M_h$. For every i such that $x_i \neq y_i$, label the geodesic that changes coordinate i last by (i, x_i). The remaining pairs (i, c) label the corresponding marker paths from the proof of Proposition E.2. Write $R^{(i,c)}$ for the path indexed by (i, c). Its penultimate horizontal vertex is

$$w^{(i,c)} := (y_1, \dots, y_{i-1}, c, y_{i+1}, \dots, y_{k-1}),$$

because its final step changes coordinate i from c to y_i.

Fix $(i, c) \in \mathcal{A}$, and write

$$R^{(i,c)} = (u_0, u_1, \dots, u_L), \qquad u_0 = \bar{x}, \quad u_L = \bar{y}, \quad u_{L-1} = w^{(i,c)}, \quad L \leq k.$$

Lifting each u_j to $u_j[q_k]$ gives a path in the top slice, although some of its internal vertices may belong to D.

If no internal vertex $u_j[q_k]$ belongs to D, use the lifted path without modification. Otherwise, let j be the smallest index such that $u_j[q_k] \in D$, and suppose that the step $u_{j-1} \to u_j$ changes horizontal axis a. Replace the missing edge into $u_j[q_k]$ by the anchor-repair edge

$$u_{j-1}[q_k] \to u_j[q_k - a],$$

and continue through the remaining horizontal changes at level $q_k - a$, stopping at $u_{L-1}[q_k - a]$.

If $u_{L-1}[q_k] \notin D$, complete the path with

$$u_{L-1}[q_k - a] \to u_{L-1}[q_k] \to u_L[q_k] = y.$$

If $u_{L-1}[q_k] \in D$, complete it instead with

$$u_{L-1}[q_k - a] \to u_{L-1}[q_k - i] \to u_L[q_k] = y.$$

The final edge in the second display exists because $u_{L-1}[q_k]$ is a deleted top-slice neighbor of y along axis i, so the construction joins y to the corresponding anchor $u_{L-1}[q_k - i]$. If two successive vertices coincide, the redundant step is omitted.

Thus, in the first case, the modified path is

$$u_0[q_k], \dots, u_{j-1}[q_k],\ u_j[q_k - a], \dots, u_{L-1}[q_k - a],\ u_{L-1}[q_k],\ u_L[q_k],$$

while in the second case $u_{L-1}[q_k]$ is replaced by $u_{L-1}[q_k - i]$. Consequently, the modification increases the length by at most one, and the resulting path has length at most $L + 1 \le k + 1$.

We now verify independence. Every internal vertex z of the modified path satisfies

$$\bar{z} \in \{u_1, \dots, u_{L-1}\},$$

so its horizontal projection is an internal vertex of the original horizontal path $R^{(i,c)}$. The two vertices $u_{L-1}[q_k - a]$ and $u_{L-1}[q_k]$, or $u_{L-1}[q_k - i]$, may have the same horizontal projection, but this causes no problem: u_{L-1} is an internal vertex belonging only to $R^{(i,c)}$.

The original horizontal paths are independent, so their sets of internal horizontal vertices are pairwise disjoint. Therefore, if two repaired paths shared an internal vertex z, then $\bar{z}$ would be an internal vertex of both corresponding horizontal paths, which is impossible. Hence the M_h lifted and repaired paths remain pairwise independent.

Moreover, they avoid the bottom slice. Every non-top level used by one of these paths has the form $q_k - a$ or $q_k - i$, which is at least 2.

Bottom-slice family. For every $c \in \{1, \dots, q_1\}$, begin with

$$x \to (c, x_2, \dots, x_{k-1}, 1).$$

If $c = x_1$, this is an ambient vertical edge; otherwise it is a bypass edge. Next change coordinates $2, \dots, k-1$, where necessary, to reach

$$(c, y_2, \dots, y_{k-1}, 1).$$

If $c = y_1$, finish with the ambient vertical edge to y; otherwise finish with the bypass edge associated with y.

Each path has length at most $1 + (k-2) + 1 = k$. There are q_1 such paths, and they are pairwise independent because every internal vertex on the path indexed by c has first coordinate c. They are independent from the horizontal family because all their internal vertices lie in the bottom slice.

Since $q_k \in \{q_1, q_1 + 1\}$, we have $q_1 \ge q_k - 1$. Therefore the two families contain at least

$$M_h + q_1 \ge M_h + (q_k - 1) = M_H$$

independent paths of length at most $k + 1$, and hence certainly at least $M_H - 1$ such paths.

Case 3: $x \in S \setminus D$ and $y \in V(\widehat{H}) \setminus S$. Write $x = \bar{x}[q_k]$, $y = \bar{y}[y_k]$ where $y_k \ne q_k$, and define

$$I := \{\ell \in \{1, \dots, k-1\} : x_\ell \ne y_\ell\}, \qquad J := \{\ell \in \{1, \dots, k-1\} : x_\ell = y_\ell\}.$$

Let $d := |I|$. We construct three families, containing respectively $M_h - d$, $q_k - 2$ and d paths.

Horizontal-marker family. For every pair (i, c) satisfying

$$\begin{cases} i \in I, \quad c \in \{1, \dots, q_i\} \setminus \{x_i, y_i\}, \\ \text{or} \\ i \in J, \quad c \in \{1, \dots, q_i\} \setminus \{x_i\}, \end{cases}$$

let $\bar{x}^{(i,c)}$ be the horizontal vector obtained from $\bar{x}$ by replacing its i-th coordinate by c.

If $\bar{x}^{(i,c)}[q_k] \notin D$, begin with the horizontal edge

$$x \to \bar{x}^{(i,c)}[q_k].$$

If $\bar{x}^{(i,c)}[q_k] \in D$, begin instead with the anchor-repair edge

$$x \to \bar{x}^{(i,c)}[q_k - i].$$

This edge exists because x is a surviving top-slice neighbor of $\bar{x}^{(i,c)}[q_k]$ along axis i.

Next move vertically to $\bar{x}^{(i,c)}[y_k]$, omitting this step if the path is already at level y_k. Change every coordinate in $I \setminus \{i\}$ from its x-value to its y-value, in any fixed order, and finally change coordinate i from c to y_i.

If $i \in I$, this path has length at most $1 + 1 + (d - 1) + 1 = d + 2 \le k + 1$. If $i \in J$, it has length at most $1 + 1 + d + 1 = d + 3 \le k + 1$, because $J \neq \emptyset$ implies $d \le k - 2$.

These paths are pairwise independent. Every internal vertex on the path indexed by (i, c) has i-th coordinate equal to

$$c \notin \{x_i, y_i\}.$$

A path indexed by (i, c'), with $c' \neq c$, has i-th coordinate c' at every internal vertex. On a path indexed by a different coordinate $i' \neq i$, coordinate i either remains unchanged or is changed directly from x_i to y_i, and therefore never takes the value c.

The number of horizontal-marker paths is

$$\sum_{i \in I} (q_i - 2) + \sum_{i \in J} (q_i - 1) = M_h - d.$$

Vertical-marker family. For each $p \in \{1, \dots, q_k - 1, q_k\} \setminus \{q_k, y_k\}$, construct a path

$$V^{(p)} : \qquad x \to \bar{x}[p] \to \cdots \to \bar{y}[p] \to y,$$

where the middle segment changes the coordinates in I from their x-values to their y-values one at a time as follows. If $d = 0$, the middle segment is empty. If $d = 1$, its order is uniquely determined. Suppose $d \ge 2$. If $p = q_k - \ell_s$ for some $s \in \{1, \dots, d\}$, change coordinate ℓ_s last. For every other value of p, use any fixed order.

There are $q_k - 2$ vertical-marker paths, each of length $d + 2 \le k + 1$. They are pairwise independent because all internal vertices of $V^{(p)}$ lie at level p, and distinct values of p give distinct levels.

Geodesic family. If $d = 0$, no paths are required in this family: the horizontal-marker and vertical-marker families already contain $M_H - 1$ paths.

Suppose that $d = 1$, so $I = \{\ell_1\}$. Define $G^{(1)}$ by

$$x \to \bar{x}[y_k] \to y.$$

Thus $G^{(1)}$ first changes the k-th coordinate and then changes coordinate ℓ_1. It has length 2, and its only internal vertex lies at level y_k, which the vertical-marker paths avoid.

Finally, suppose that $d \geq 2$, and write $I = \{\ell_1 < \cdots < \ell_d\}$. For each $s \in \{1, \ldots, d\}$, let

$$\bar{x}^{(s)} := (x_1, \ldots, x_{\ell_s - 1}, y_{\ell_s}, x_{\ell_s + 1}, \ldots, x_{k-1}).$$

If $\bar{x}^{(s)}[q_k] \notin D$, begin $G^{(s)}$ with

$$x \to \bar{x}^{(s)}[q_k].$$

If $\bar{x}^{(s)}[q_k] \in D$, begin instead with the anchor-repair edge

$$x \to \bar{x}^{(s)}[q_k - \ell_s].$$

Next change the k-th coordinate to y_k, omitting this step if the path is already at that level, and then change the remaining horizontal coordinates in the cyclic order

$$\ell_{s+1}, \ldots, \ell_d, \ell_1, \ldots, \ell_{s-1}.$$

This gives d paths, each of length at most $d + 1 \leq k$.

We claim that the paths $G^{(1)}, \ldots, G^{(d)}$ are pairwise independent. For $s = 1, \ldots, d$, let

$$\rho^{(s)} := (\ell_s, \ell_{s+1}, \ldots, \ell_d, \ell_1, \ldots, \ell_{s-1}),$$

and define

$$F(z) := \{\ell \in I : z_\ell = y_\ell\}.$$

Along $G^{(s)}$, the horizontal coordinates are changed in the order $\rho^{(s)}$. Consequently, at every internal vertex z of $G^{(s)}$,

$$F(z) = \{\rho^{(s)}(1), \ldots, \rho^{(s)}(r)\}$$

for some $r \in \{1, \ldots, d-1\}$. The same set may occur at two consecutive vertices of $G^{(s)}$, because the vertical step does not change any horizontal coordinate. Likewise, an anchor repair changes the level of the first corrected horizontal vector but does not change the corresponding set $F(z)$.

Suppose that $G^{(s)}$ and $G^{(s')}$ share an internal vertex z. Then

$$\{\rho^{(s)}(1), \ldots, \rho^{(s)}(r)\} = F(z) = \{\rho^{(s')}(1), \ldots, \rho^{(s')}(r')\}$$

for some $r, r' \in \{1, \ldots, d-1\}$. Equality of the cardinalities gives $r = r'$, and equality of the two cyclic initial segments then gives $s = s'$, exactly as in the proof of Proposition 5.10. Thus two distinct paths cannot share an internal vertex, proving independence.

It remains to verify independence between the three families. The horizontal-marker paths are independent from the other two families because they carry a horizontal coordinate value outside the corresponding endpoint values. The geodesic and vertical-marker paths use only x- and y-values in every horizontal coordinate.

When $d = 1$, the only internal vertex of $G^{(1)}$ lies at level y_k, which the vertical-marker paths avoid. Suppose therefore that $d \geq 2$, and fix $s \in \{1, \ldots, d\}$. Apart from a possible anchor vertex, all internal vertices of $G^{(s)}$ lie either in the top slice or at level y_k, and both levels are avoided by the vertical-marker paths.

If the first intended top-slice vertex of $G^{(s)}$ is deleted, the path uses an anchor at level $q_k - \ell_s$. If $q_k - \ell_s = y_k$, this level is excluded from the vertical-marker family. Otherwise, the only vertical-marker path at the anchor level is $V^{(q_k - \ell_s)}$. This path changes coordinate ℓ_s last. Before doing so, coordinate ℓ_s still has its x-value; afterwards, every coordinate in I has its y-value. Since $d \geq 2$, it never visits the vertex at which only coordinate ℓ_s has been changed to its y-value. Thus it does not meet the anchor vertex of $G^{(s)}$.

Hence all three families are mutually independent. Their total number of paths is

$$(M_h - d) + (q_k - 2) + d = M_h + (q_k - 2) = M_H - 1.$$

Every path has length at most $k + 1$, completing Case 3.

Thus every two distinct vertices of $\widehat{H}$ are joined by $M_H - 1$ independent paths of length at most $k + 1$. Therefore $\widehat{H}$ is $(k+1)$-step $(M_H - 1)$-connected. $\square$

Proof of Theorem 5.19. By Theorem 5.18(ii), $\widehat{H}$ is $(k+1)$-step (M_H-1)-connected, and so

$$\min\Big\{\|G\| : |G| = n,\ G \text{ is } (k+1)\text{-step } (M_H-1)\text{-connected}\Big\} \le \|\widehat{H}\|.$$

On the other hand, every $(k+1)$-step (M_H-1)-connected graph has minimum degree at least $M_H - 1$. Therefore the handshake lemma gives

$$\min\Big\{\|G\| : |G| = n,\ G \text{ is } (k+1)\text{-step } (M_H-1)\text{-connected}\Big\} \ge \frac{n(M_H-1)}{2}.$$

It remains to bound $\|\widehat{H}\|$ from above. Set $Q := \prod_{\ell=1}^{k} q_\ell$, $N := \prod_{\ell=1}^{k-1} q_\ell$ and $t := Q - n$. Thus $Q = n + t$, $N = |S|$, and $t = |D|$. Let $e_S(D)$ denote the number of top-slice edges with both endpoints in D.

We begin with the ambient Hamming graph $H(q_1, \dots, q_k)$, which has

$$\frac{QM_H}{2} = \frac{(n+t)M_H}{2}$$

edges. Deleting the t vertices of D removes

$$tM_H - e_S(D)$$

edges. The term tM_H counts all edges incident to vertices in D, but each edge with both endpoints in D is counted twice, so we subtract $e_S(D)$.

Next, we add the anchor-repair edges. These repair exactly the deleted edges from D to $S \setminus D$ inside the top slice. Each deleted vertex has $\sum_{\ell=1}^{k-1}(q_\ell - 1) = M_H - (q_k - 1)$ neighbors in the top slice, so altogether the deleted vertices are incident to $t(M_H - (q_k - 1))$ top-slice edges. However, this counts each edge with both endpoints in D twice. Hence the number of deleted top-slice edges from D to $S \setminus D$ is

$$t\big(M_H - (q_k - 1)\big) - 2e_S(D).$$

These are precisely the number of anchor-repair edges that are added back.

Finally, the top-to-bottom bypass step adds

$$(N - t)(q_1 - 1)$$

edges, since there are $N - t$ surviving top-slice vertices and each receives exactly $q_1 - 1$ bypass edges to the bottom slice.

Putting everything together,

$$\begin{aligned}\|\widehat{H}\| &= \frac{(n+t)M_H}{2} - \big(tM_H - e_S(D)\big) + \Big(t\big(M_H - (q_k-1)\big) - 2e_S(D)\Big) + (N-t)(q_1-1)\\ &= \frac{(n+t)M_H}{2} - t(q_k-1) - e_S(D) + (N-t)(q_1-1).\end{aligned}$$

Since $e_S(D) \ge 0$, $t \ge 0$ and $q_1 \le q_k$, we obtain

$$\|\widehat{H}\| \le \frac{nM_H}{2} + t\Big(\frac{M_H}{2} - (q_k-1)\Big) + N(q_1-1) \le \frac{nM_H}{2} + t\Big(\frac{k}{2} - 1\Big)(q_k-1) + N(q_k-1),$$

because $M_H \le k(q_k - 1)$. Since $t < N$, this yields

$$\|\widehat{H}\| \le \frac{nM_H}{2} + \frac{k}{2}N(q_k-1).$$

Finally, from $n = Nq_k - t$ and $t < N$, we have $n > N(q_k - 1)$, and therefore

$$\|\widehat{H}\| \le \frac{nM_H}{2} + \frac{nk}{2} = \frac{n(M_H+k)}{2}. \qquad \square$$

Proof of Theorem 5.20. For part (i), assume $n > k^k$. Let $\widehat{H} \in \mathrm{Hydra}(n,k)$, embedded in an ambient Hamming graph $H(q_1,\dots,q_k)$, with top slice S, deletion set $D \subset S$, and surviving reference slice $T = S \setminus D \neq \emptyset$. Let a be the ambient position vacated by the deletion.

If $a \notin T$, Algorithm 3 chooses a position $b \in T$ and relocates the surviving vertex at b to a. The vacant position is then b. Relabelling this new vacant position as a, we may therefore assume that $a \in T$.

If $T \setminus \{a\} \neq \emptyset$, then adding the anchor-repair edges associated with the deletion of a gives exactly the Hydra construction on $n-1$ vertices with the same ambient Hamming graph. Hence the output lies in $\mathrm{Hydra}(n-1,k)$.

If $T \setminus \{a\} = \emptyset$, then $T = \{a\}$, so $|D| = |S| - 1$. After the deletion, the whole top slice is vacant. Algorithm 3 therefore passes to the ambient Hamming graph obtained by replacing q_k by $q_k - 1$ and reordering the side lengths so that they are as balanced as possible for $(n-1,k)$. In this new ambient graph the deletion set is empty, and the required bypass edges give exactly a graph in $\mathrm{Hydra}(n-1,k)$.

For part (ii), let $\widehat{H} \in \mathrm{Hydra}(n,k)$. If $D \neq \emptyset$, then Algorithm 4 fills one vacant top-slice position $v \in D$, removes the associated anchor-repair edges, restores the ambient Hamming adjacencies at v, and replaces D by $D \setminus \{v\}$. This is exactly the Hydra construction on $n+1$ vertices with the same ambient Hamming graph, so the output lies in $\mathrm{Hydra}(n+1,k)$.

If $D = \emptyset$, then the current graph already fills its ambient Hamming graph. Algorithm 4 therefore passes to the reordered side lengths $(q_2,\dots,q_k,q_1+1)$, which are as balanced as possible for $(n+1,k)$. Choosing one vertex v in the new top slice and setting $D' = S' \setminus \{v\}$, together with the corresponding anchor-repair and bypass edges, reproduces the Hydra construction on $n+1$ vertices. Hence the output lies in $\mathrm{Hydra}(n+1,k)$. □

Proof of Proposition 6.1. For every feasible dimension r, let $M_{n,r} := \sum_{\ell=1}^{r}(q_\ell - 1)$, where $q_1,\dots,q_r$ are as balanced as possible for (n,r), and define the function $g_n(t) := tn^{1/t}$ for $t > 0$. Since $q_\ell \in \{\lfloor n^{1/r}\rfloor, \lfloor n^{1/r}\rfloor + 1\}$, we have

$$g_n(r) - 2r \le M_{n,r} \le g_n(r). \tag{E.1}$$

We first prove the edge-minimization claim. Fix any feasible integer $k < K_n$. The derivative of g_n satisfies

$$-g_n'(t) = n^{1/t}\left(\frac{\log n}{t} - 1\right).$$

For every $t \in [k, K_n]$, the definition of K_n gives $n^{1/t} \ge n^{1/K_n} \ge K_n$ and $(\log n)/t - 1 \ge (\log n)/K_n - 1 \ge \log K_n - 1$. Consequently,

$$-g_n'(t) \ge K_n(\log K_n - 1), \qquad t \in [k, K_n].$$

Integrating over this interval yields

$$g_n(k) - g_n(K_n) = \int_k^{K_n} -g_n'(t)\,dt \ge (K_n - k)K_n(\log K_n - 1). \tag{E.2}$$

Combining (E.1) and (E.2), we obtain

$$\begin{aligned}(M_{n,k} - 1) - (M_{n,K_n} + K_n) &\ge g_n(k) - g_n(K_n) - 2k - K_n - 1\\ &\ge (K_n - k)K_n(\log K_n - 1) - 2k - K_n - 1\\ &\ge K_n(\log K_n - 4) - 1.\end{aligned}$$

Since $K_n \to \infty$, the final expression is strictly positive for all sufficiently large n, uniformly over every feasible $k < K_n$. Now let $\widehat{H} \in \mathrm{Hydra}(n,k)$ and $\widehat{H}' \in \mathrm{Hydra}(n,K_n)$. Theorem 5.19 therefore gives, for all sufficiently large n,

$$\|\widehat{H}\| \ge \frac{n(M_{n,k} - 1)}{2} > \frac{n(M_{n,K_n} + K_n)}{2} \ge \|\widehat{H}'\|.$$

Thus every Hydra graph of dimension K_n has fewer edges than every Hydra graph of any lower feasible dimension. Since no dimension greater than K_n is feasible, K_n is the unique edge-minimizing dimension for all sufficiently large n.

It remains to determine the resulting asymptotic edge count. Let W denote the principal branch of the Lambert W-function. By definition, K_n is the largest integer satisfying $K_n^{K_n} \le n$. The real solution of $x \log x = \log n$ is $x = \log n / W(\log n)$, and therefore

$$K_n = \Big\lfloor \frac{\log n}{W(\log n)} \Big\rfloor.$$

Since $W(\log n) \sim \log \log n$, it follows that

$$K_n \sim \frac{\log n}{\log \log n}.$$

Now set $k := K_n$. Since $k^k \le n < (k+1)^{k+1}$, we have $k \le n^{1/k} < (k+1)^{1+1/k}$. Hence $n^{1/k} = k + O(\log k)$, and so each balanced side length satisfies $q_\ell = n^{1/k} + O(1) = k + O(\log k)$. Therefore

$$M_n = \sum_{\ell=1}^{k} (q_\ell - 1) = k^2 + O(k \log k).$$

Finally, Theorem 5.19 gives

$$\frac{n(M_n - 1)}{2} \le \|\widehat{H}_n\| \le \frac{n(M_n + k)}{2}.$$

Substituting $M_n = k^2 + O(k \log k)$ yields

$$\|\widehat{H}_n\| = \frac{n}{2}\big(k^2 + O(k \log k)\big) = \left(\frac{1}{2} + o(1)\right) n k^2.$$

Using $k = K_n \sim \log n / \log \log n$ proves the claim. □

Proof of Proposition 6.2. Write $m_{n,k} := k(\lfloor n^{1/k} \rfloor - 1) - 1$ and $A_k(p) := 1 - (1-p)^{k+1}$, so

$$\bar{\pi}_{n,k}(p) = A_k(p)^{m_{n,k}}.$$

For fixed k, the quantity $\lfloor n^{1/k} \rfloor$ is nondecreasing in n. Hence $m_{n,k}$ is nondecreasing in n. Since the base $A_k(p)$ lies strictly between zero and one, it follows that $\bar{\pi}_{n,k}(p)$ is nonincreasing in n. This proves (i).

We next fix n and compare two adjacent feasible dimensions k and $k+1$. Set $x := n^{1/(k+1)}$. Since $k+1$ is feasible, $x \ge k+1$. We claim that

$$m_{n,k} > m_{n,k+1}. \tag{E.3}$$

Indeed, $m_{n,k} + 1 = k(\lfloor n^{1/k} \rfloor - 1) \ge k x^{1+1/k} - 2k$, whereas $m_{n,k+1} + 1 = (k+1)(\lfloor x \rfloor - 1) \le (k+1)(x-1)$. It is therefore enough to show that

$$\psi_k(x) := k x^{1+1/k} - (k+1)x - k + 1 > 0$$

whenever $x \ge k+1$ and $k \ge 3$. Since

$$\psi_k'(x) = (k+1)\big(x^{1/k} - 1\big) > 0,$$

it suffices to evaluate ψ_k at $x = k+1$.

For $k=3$, $\psi_3(4)=12\sqrt[3]{4}-18>0$. For $k\geq 4$, the inequality $e^y\geq 1+y$ gives

$$\psi_k(k+1)=k(k+1)\left((k+1)^{1/k}-1\right)-2k\geq (k+1)\log(k+1)-2k>0.$$

The final inequality holds at $k=4$ and its left-hand side is increasing for $k\geq 4$. This proves (E.3). Combining this with $0<A_k(p)<A_{k+1}(p)<1$ allows us to obtain

$$\bar{\pi}_{n,k+1}(p)=A_{k+1}(p)^{m_{n,k+1}}>A_k(p)^{m_{n,k+1}}>A_k(p)^{m_{n,k}}=\bar{\pi}_{n,k}(p).$$

Thus $\bar{\pi}_{n,k}(p)$ is strictly increasing in k over the feasible dimensions. In particular,

$$\min_{3\leq k\leq K_n}\bar{\pi}_{n,k}(p)=\bar{\pi}_{n,3}(p). \qquad \square$$

Proof of Theorem 6.4. Write $a:=\lambda u$ and $b:=\lambda v$. We first study the asymptotic contribution of the failure-loss term when k is fixed. Let $A_k(p):=1-(1-p)^{k+1}\in(0,1)$. Since $m_{n,k}=k(\lfloor n^{1/k}\rfloor-1)-1=kn^{1/k}+O(1)$, we have

$$\bar{\pi}_{n,k}(p)=A_k(p)^{m_{n,k}}=\exp\left\{-\gamma_k(p)n^{1/k}+O(1)\right\},$$

where $\gamma_k(p):=-k\log A_k(p)>0$. It follows that, for every fixed k and every $r>0$,

$$n^r\bar{\pi}_{n,k}(p)\to 0. \tag{E.4}$$

We next show that when $\alpha>1$, every sequence of minimizers is bounded. Choose a fixed integer $k_0\geq 3$ such that $1+1/k_0\leq\alpha$. For all sufficiently large n, the value k_0 is feasible, and

$$C_\alpha(k_0;n,\lambda,u,v,p)\leq\left(\frac{k_0}{2}+ak_0+b\right)n^\alpha.$$

On the other hand, all three cost terms are nonnegative, so $C_\alpha(k_n^\star;n,\lambda,u,v,p)\geq ak_n^\star n^\alpha$. The optimality of $k_n^\star$ therefore gives

$$k_n^\star\leq\frac{k_0/2+ak_0+b}{a}.$$

Thus, when $\alpha>1$, it is sufficient to compare a fixed finite collection of dimensions.

(i) Suppose first that $\alpha>4/3$. Equation (E.4) together with the fact $1+1/k\leq 4/3<\alpha$ for every fixed $k\geq 3$, consequently give $C_\alpha(k;n,\lambda,u,v,p)/n^\alpha\to ak$. This limit is uniquely minimized over $k\geq 3$ by $k=3$.

(ii) Now suppose that $1<\alpha<4/3$ and set $r_\alpha:=1/(\alpha-1)$ and $q_\alpha:=\lceil r_\alpha\rceil$. If $r_\alpha\notin\mathbb{N}$, then $q_\alpha-1<r_\alpha<q_\alpha$. For every fixed $k<q_\alpha$, the maintenance exponent $1+1/k>\alpha$, and hence $C_\alpha(k;n,\lambda,u,v,p)/n^\alpha\to\infty$. For every fixed $k\geq q_\alpha$, the maintenance exponent $1+1/k<\alpha$, so that $C_\alpha(k;n,\lambda,u,v,p)/n^\alpha\to ak$. Among these dimensions, the limiting coefficient is uniquely minimized at $k=q_\alpha$.

(iii) It remains to consider the critical exponents $\alpha=1+1/j$, where $j\geq 3$ is an integer. For $k<j$, the maintenance term has order strictly greater than n^α, so such dimensions cannot be optimal. At $k=j$, $C_\alpha(j;n,\lambda,u,v,p)/n^\alpha\to aj+\frac{j}{7}2$. For $k>j$, $C_\alpha(k;n,\lambda,u,v,p)/n^\alpha\to ak$, whose smallest value is attained at $k=j+1$. The limiting comparison is therefore between $aj+j/2$ and $a(j+1)$. Thus $j+1$ is optimal when $a<j/2$, while j is optimal when $a>j/2$.

Suppose finally that $a=j/2$, so that the leading coefficients coincide. Since $\alpha=1+1/j$, the maintenance cost at $k=j$ is exactly cancelled by the additional usage cost incurred when passing from j to $j+1$. Hence

$$\begin{aligned}C_\alpha(j+1;n,\lambda,u,v,p)-C_\alpha(j;n,\lambda,u,v,p)&=\frac{j+1}{2}n^{1+1/(j+1)}+bn^\alpha\left(\bar{\pi}_{n,j+1}(p)-\bar{\pi}_{n,j}(p)\right)\\&=\frac{j+1}{2}n^{1+1/(j+1)}+o(1).\end{aligned}$$

The first term tends to infinity. Hence the difference is eventually positive, and j is optimal.

(iv) Finally, suppose that $\alpha = 1$. Extend the maintenance factor $kn^{1/k}$ from integer dimensions to the function $g_n(t) := tn^{1/t}$ for $t > 0$. Its derivative satisfies

$$-g_n'(t) = n^{1/t}\left(\frac{\log n}{t} - 1\right).$$

Now fix any feasible integer $k < K_n$. For every $t \in [k, K_n]$, the definition of K_n gives $n^{1/t} \geq n^{1/K_n} \geq K_n$ and $(\log n)/t - 1 \geq (\log n)/K_n - 1 \geq \log K_n - 1$. Consequently,

$$-g_n'(t) \geq K_n(\log K_n - 1), \qquad t \in [k, K_n].$$

Integrating over this interval yields

$$g_n(k) - g_n(K_n) = \int_k^{K_n} -g_n'(t)\,dt \geq (K_n - k)K_n(\log K_n - 1).$$

Since $0 \leq \bar{\pi}_{n,k}(p) \leq 1$, it follows that

$$\begin{aligned} &C_1(k; n, \lambda, u, v, p) - C_1(K_n; n, \lambda, u, v, p) \\ &\qquad = \frac{n}{2}(g_n(k) - g_n(K_n)) - an(K_n - k) + bn(\bar{\pi}_{n,k}(p) - \bar{\pi}_{n,K_n}(p)) \\ &\qquad \geq n\left[(K_n - k)\left(\frac{K_n}{2}(\log K_n - 1) - a\right) - b\right]. \end{aligned}$$

Now $K_n(\log K_n - 1)/2 - a \to \infty$. Since $K_n - k \geq 1$, the preceding lower bound is strictly positive, uniformly over all feasible $k < K_n$, for all sufficiently large n. Therefore $k_n^\star = K_n$ for all sufficiently large n. □

References

R. Albert, H. Jeong, and A. L. Barabási. Error and attack tolerance of complex networks. *nature*, 406(6794):378–382, 2000.

Amboss Technologies. Lightning network statistics & analytics. `https://amboss.space/stats`, 2026. Accessed 11 August 2026.

Yackolley Amoussou-Guenou, Bruno Biais, Maria Potop-Butucaru, and Sara Tucci-Piergiovanni. Rationals vs byzantines in consensus-based blockchains, 2019a.

Yackolley Amoussou-Guenou, Antonella del Pozzo, Maria Potop-Butucaru, and Sara Tucci-Piergiovanni. On fairness in committee-based blockchains, 2019b.

A. L. Barabási and R. Albert. Emergence of scaling in random networks. *science*, 286(5439): 509–512, 1999.

D. Belotserkovsky. The smallest number of edges in a 2-connected graph with specified diameter. *Discrete mathematics*, 307(19-20):2376–2384, 2007.

Bhuyan and Agrawal. Generalized hypercube and hyperbus structures for a computer network. *IEEE Transactions on computers*, 100(4):323–333, 1984.

Bruno Biais, Christophe Bisière, Matthieu Bouvard, and Catherine Casamatta. The blockchain folk theorem. *The Review of Financial Studies*, 32(5):1662–1715, 2019. doi: 10.1093/rfs/hhy095.

B. Bollobás. *Extremal graph theory*. Courier Corporation, 2004.

B. Bollobás and P. Erdös. An extremal problem of graphs with diameter 2. *Mathematics Magazine*, 48(5):281–283, 1975.

B. Bollobás and F. Harary. Extremal graphs with given diameter and connectivity. *Ars Combinatoria*, 1:281–296, 1976.

Andries E. Brouwer and Willem H. Haemers. *Spectra of Graphs*. Springer, New York, 2012. doi: 10.1007/978-1-4614-1939-6.

L. Caccetta. On extremal graphs with given diameter and connectivity. *Annals of the New York Academy of Sciences*, 328(1):76–94, 1979.

F. R. K. Chung. Graphs with small diameter after edge deletion. *Discrete applied mathematics*, 37:73–94, 1992.

F. R. K. Chung and M. R. Garey. Diameter bounds for altered graphs. *Journal of graph theory*, 8(4):511–534, 1984.

R. Diestel. *Graph theory*, volume 173. Springer Nature, sixth edition, 2025.

Ethereum Foundation. Ethereum consensus specifications: Phase 0 p2p interface. `github.com/ethereum/consensus-specs`, 2026. Phase 0 p2p-interface specification, accessed May 30, 2026.

R. J. Faudree and Z. Tuza. Stronger bounds for generalized degrees and menger path systems. *Discussiones Mathematicae Graph Theory*, 15(2):167–177, 1995.

R. J. Faudree, M. S. Jacobson, E. T. Ordman, R. Schelp, and Z. Tuza. Menger's theorem and short paths. *Journal of Combinatorial Mathematics and Combinatorial Computing*, 2: 235–253, 1987.

R. J. Faudree, R. J. Gould, and H. R. Schelp. Menger path systems. *Journal of Combinatorial Mathematics and Combinatorial Computing*, 6:9–21, 1989.

R. J. Faudree, R. J. Gould, and L. M. Lesniak. Generalized degrees and menger path systems. *Discrete applied mathematics*, 37:179–191, 1992.

R. J. Faudree, R. J. Gould, and J.S. Powell. Property pd, m and efficient design of reliable networks. *Networks*, 60(3):167–178, 2012.

E. Flandrin and H. Li. Mengerian properties, hamiltonicity, and claw-free graphs. *Networks*, 24(3):177–183, 1994.

K. I. Goh, B. Kahng, and D. Kim. Universal behavior of load distribution in scale-free networks. *Physical review letters*, 87(27):278701, 2001.

Paolo Guasoni and Nazem Khan. Chain or channel? payment optimization with heterogeneous flow. *Available at SSRN: https://ssrn.com/abstract=5046421*, 2024.

Paolo Guasoni, Gur Huberman, and Clara Shikhelman. Lightning network economics: Channels. *Management Science*, 70(6):3827–3840, 2024.

Paolo Guasoni, Gur Huberman, and Clara Shikhelman. Lightning network economics: Topology. *Management Science*, 71(7):5477–5490, 2025.

F. Harary. The maximum connectivity of a graph. *Proceedings of the National Academy of Sciences*, 48(7):1142–1146, 1962.

Frank Harary, John P Hayes, and Horng-Jyh Wu. A survey of the theory of hypercube graphs. *Computers & Mathematics with Applications*, 15(4):277–289, 1988.

Friedrich A. Hayek. The use of knowledge in society. *The American Economic Review*, 35(4): 519–530, September 1945. URL `https://www.jstor.org/stable/1809376`.

Thomas P Hayes, Jared Saia, and Amitabh Trehan. The forgiving graph: a distributed data structure for low stretch under adversarial attack. In *Proceedings of the 28th ACM symposium on Principles of distributed computing*, pages 121–130, 2009.

Shlomo Hoory, Nathan Linial, and Avi Wigderson. Expander graphs and their applications. *Bulletin of the American Mathematical Society*, 43(4):439–561, 2006. doi: 10.1090/S0273-0979-06-01126-8.

D. F. Hsu. On container width and length in graphs, groups, and networks–dedicated to professor paul erdös on the occasion of his 80th birthday–. *IEICE transactions on fundamentals of electronics, communications and computer sciences*, 77(4):668–680, 1994.

Hervé Kerivin and A Ridha Mahjoub. Design of survivable networks: A survey. *Networks: An International Journal*, 46(1):1–21, 2005.

Kimberly Lange, Elias Rohrer, and Florian Tschorsch. On the impact of attachment strategies for payment channel networks. In *2021 IEEE International Conference on Blockchain and Cryptocurrency (ICBC)*, pages 1–9. IEEE, 2021.

Alfred Lehar, Christine A. Parlour, and Marius Zoican. Fragmentation and optimal liquidity supply on decentralized exchanges, 2023.

Lightning Labs. Lightning service provider. `https://docs.lightning.engineering/the-lightning-network/liquidity/lightning-service-provider`, 2022. Accessed 11 August 2026.

Jian-Hong Lin, Kevin Primicerio, Tiziano Squartini, Christian Decker, and Claudio J Tessone. Lightning network: a second path towards centralisation of the bitcoin economy. *New Journal of Physics*, 22(8):083022, 2020.

L. Lovász, V. Neumann-Lara, and M. Plummer. Mengerian theorems for paths of bounded length. *Periodica Mathematica Hungarica*, 9(4):269–276, 1978.

Yajun Lu, Hosseinali Salemi, Balabhaskar Balasundaram, and Austin Buchanan. On fault-tolerant low-diameter clusters in graphs. *INFORMS Journal on Computing*, 34(6):3181–3199, 2022.

Thomas L Magnanti and Richard T Wong. Network design and transportation planning: Models and algorithms. *Transportation science*, 18(1):1–55, 1984.

Mohammad Saleh Mahdizadeh, Behnam Bahrak, and Mohammad Sayad Haghighi. Decentralizing the lightning network: a score-based recommendation strategy for the autopilot system. *Applied Network Science*, 8(1):73, 2023.

Stefano Martinazzi and Andrea Flori. The evolving topology of the lightning network: Centralization, efficiency, robustness, synchronization, and anonymity. *Plos one*, 15(1):e0225966, 2020.

K. Menger. Zur allgemeinen kurventheorie. *Fundamenta Mathematicae*, 10(1):96–115, 1927.

Fred J. Meyer. Flip-trees: Fault-tolerant graphs with wide containers. *IEEE Transactions on Computers*, 37(4):472–478, 2002.

M. Miller and J. Širáň. Moore graphs and beyond: A survey of the degree/diameter problem. *The electronic journal of combinatorics*, page DS14, 2013.

A. E. Motter and Y. C. Lai. Cascade-based attacks on complex networks. *Physical Review E*, 66(6):065102, 2002.

U. S. R. Murty. On critical graphs of diameter 2. *Mathematics Magazine*, 41(3):138–140, 1968.

U. S. R. Murty and K. Vijayan. On accessibility in graphs. *Sankhyā: The Indian Journal of Statistics, Series A*, pages 299–302, 1964.

Young-Soo Myung, Hyun-joon Kim, and Dong-wan Tcha. Design of communication networks with survivability constraints. *Management Science*, 45(2):238–252, 1999.

Gleb Naumenko, Gregory Maxwell, Pieter Wuille, Alexandra Fedorova, and Ivan Beschastnikh. Bandwidth-efficient transaction relay for bitcoin, 2019.

Gopal Pandurangan, Peter Robinson, and Amitabh Trehan. Dex: self-healing expanders. *Distributed Computing*, 29(3):163–185, 2016.

Joseph Poon and Thaddeus Dryja. The bitcoin lightning network: Scalable off-chain instant payments, 2016.

L. Pyber and Z. Tuza. Menger-type theorems with restrictions on path lengths. *Discrete Mathematics*, 120(1-3):161–174, 1993.

Jared Saia and Amitabh Trehan. Picking up the pieces: Self-healing in reconfigurable networks. In *2008 IEEE International Symposium on Parallel and Distributed Processing*, pages 1–12. IEEE, 2008.

Fahad Saleh. Blockchain without waste: Proof-of-stake. *The Review of Financial Studies*, 34 (3):1156–1190, 2021. doi: 10.1093/rfs/hhaa075.

Sindura Saraswathi and Christian Kümmerle. An exposition of pathfinding strategies within lightning network clients. In *2025 IEEE International Conference on Blockchain and Cryptocurrency (ICBC)*, pages 1–17. IEEE, 2025.

István András Seres, László Gulyás, Dániel A Nagy, and Péter Burcsi. Topological analysis of bitcoin's lightning network. In *Mathematical Research for Blockchain Economy: 1st International Conference MARBLE 2019, Santorini, Greece*, pages 1–12. Springer, 2020.

Dimitris Vyzovitis, Yusef Napora, Dirk McCormick, David Dias, and Yiannis Psaras. Gossipsub: Attack-resilient message propagation in the filecoin and eth2.0 networks, 2020.

J. Xu. *Topological structure and analysis of interconnection networks*, volume 7. Springer Science & Business Media, 2001.

Philipp Zabka, Klaus-T Foerster, Stefan Schmid, and Christian Decker. Empirical evaluation of nodes and channels of the lightning network. *Pervasive and Mobile Computing*, 83:101584, 2022.

Marius Zoican and Sorin Zoican. Fragmentation of distributed exchanges, 2019.