Resilient AI Infrastructure by Design: A Spatially-Aware Framework for Tolerating Clustered Failures

Yiquan Wang,[1] Ziyang Liu,[2] Jingfan Zai,[1] Jingyi Chen,[1] Eminjan Sabir[1]*

[1]College of Mathematics and System Sciences, Xinjiang University, Urumqi 830046, China
[2]School of Future Science and Engineering, Soochow University, Suzhou 215000, China
eminjan20150513@163.com

Abstract

Training large-scale AI models is a massive investment, yet these multi-million dollar runs are extraordinarily vulnerable to physical infrastructure failures. A single component failure, like a rack power supply, can trigger a cascade of spatially correlated, clustered network failures, catastrophically terminating the entire task. This fragility presents a critical challenge to the reliable deployment of AI in the real world, stemming from a fundamental flaw in system design: abstract fault models are blind to the physical reality of failures. They cannot capture spatial correlation, leading to systems that are either over-provisioned or deceptively brittle. We address this challenge by proposing the Region-Based Fault (RBF) framework, a new paradigm in computing systems design for AI that treats the physical topology of failures as a first-order parameter. We formally prove a key principle for resilient system design: the geometric dispersion of faults is a more critical determinant of network resilience than their aggregate count. We prove that strategically increasing the physical separation (`dsep`) between potential fault regions provides disproportionately high gains in resilience. We prove that the k-ary n-cube, a prevalent AI interconnect topology, maintains Hamiltonian connectivity—a property essential for high-performance, deadlock-free communication—under a wide range of realistic, clustered fault conditions. This work provides system architects with a mathematical foundation and actionable algorithms to design next-generation AI infrastructures that are not only more resilient to real-world failures but also more cost-effective, directly tackling a fundamental deployment challenge and paving the way for more reliable and accelerated AI-driven discoveries.

Keywords: Region-Based Fault Model (RBF), Spatially-Aware Fault Tolerance, AI Infrastructure, k-ary n-cubes, Hamiltonian Connectivity.

Introduction

The success of large-scale AI, from foundation models to scientific discovery, is built on computation at scale, with systems integrating tens of thousands of processors (Jouppi et al. 2017, 2023; NVIDIA 2025). The k-ary n-cube (Q_n^k) topology, with its scalable and regular structure, remains a cornerstone of these architectures, from Google's TPU Pods to specialized accelerators (Dally 2002; Sarbazi-Azad, Ould-Khaoua, and Mackenzie 2001; Qiao and Zhang 2024). However, the scale and complexity that enable AI breakthroughs also render these systems fragile to physical failures. This operational fragility has become a key challenge in applying and deploying AI in the real world, where stability over weeks or months is required.

The root of this challenge lies in a disconnect between the physical reality of failures and the abstract models used for system design. In large-scale infrastructure, failures are rarely random. A single physical event, like a cooling unit malfunction, can create a spatially correlated, clustered "hole" in the network by disabling an entire rack (Hacker, Romero, and Carothers 2009; Vishwanath and Nagappan 2010; Pinheiro, Weber, and Barroso 2007; Xia et al. 2020; Chen, Chao, and Wu 2015). Such an event can trigger communication deadlocks, crashing expensive training jobs and compromising network properties like Hamiltonian connectivity, which is important for deadlock-free routing (Dean et al. 2012; Ali, Abbas, and Khan 2004; Dong et al. 2025). However, prevailing fault models are blind to this physical reality. The classic Fault-Tolerant (FT) model's assumption of independent, arbitrary faults ignores the primary risk of clustered events (Yang, Tan, and Hsu 2007; Ashir and Stewart 2002; Lv et al. 2018). Even more advanced models, while accounting for logical dimensions, remain blind to physical locality, misinterpreting a single physical failure as multiple unrelated faults (Zhuang et al. 2023b, 2024, 2023a). This disconnect leaves system architects designing in the dark, unable to build systems resilient against probable and damaging real-world failure scenarios.

To address this deployment challenge, we introduce the Region-Based Fault (RBF) framework, a contribution to the design and optimization of computing systems for AI. RBF is a hardware-software co-design paradigm that explicitly incorporates the spatial geometry of the physical system into network analysis by modeling failures as connected "fault clusters" with parameters for size, shape, and physical separation (`dsep`). In this context, we formally prove that fault dispersion is a more critical determinant of resilience than aggregate fault count, showing that increasing `dsep` yields super-linear gains. We provide a constructive, spatially-aware algorithm that guarantees Hamiltonian connectivity in Q_n^k networks and validate its performance ex-

*Corresponding author.

perimentally, demonstrating its robustness far beyond theoretical guarantees.

Related Work

Research on fault tolerance in large-scale interconnection networks has evolved toward more realistic failure models. The traditional Fault-Tolerant (FT) model provides worst-case guarantees by assuming faults can be independent and arbitrarily placed. While foundational, its random failure assumption leads to overly pessimistic assessments, ill-suited for real-world, clustered events like rack-level failures (Yang, Tan, and Hsu 2007; Ashir and Stewart 2002). For instance, the FT model limits the fault tolerance of a k-ary n-cube to $2n-3$ edges, a bound dictated by the pathological (and improbable) scenario of all faults targeting a single node.

To overcome this limitation, the Partitioned Edge Fault (PEF) model marked a significant step forward by distinguishing failures across logical dimensions (Zhuang et al. 2023b). This insight, motivated by architectures where dimensional links have different failure characteristics, dramatically improved the theoretical fault tolerance bounds from linear to exponential in the network's dimension. However, the PEF model's critical flaw lies in its blindness to physical proximity. A single physical event—such as a localized cooling failure—can sever links across multiple logical dimensions, which PEF misinterprets as multiple independent faults. This reveals a critical gap in existing models: their inability to capture the "physical geometry" of failures. Our Region-Based Fault (RBF) model fills this void by treating the spatial distribution and clustering of failures as first-class citizens, providing a framework to realistically evaluate and design highly resilient AI infrastructure.

While other advanced conditional and structure-based models have been proposed to avoid the FT model's worst-case scenarios, they often focus on the properties of the surviving graph rather than the topology of the fault region itself. For a detailed survey of classical and conditional fault models, including specific theoretical bounds and a comparative analysis, please refer to Appendix A.

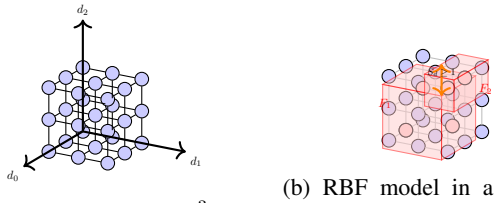
Preliminaries

Graph-Theoretic Notation. We model the network as a graph $G = (V, E)$. A network with link failures $F \subseteq E$ is denoted as the faulty graph $G - F$. A path is a sequence of distinct, connected vertices. A *Hamiltonian path* visits every vertex in the graph exactly once, and a graph is *Hamiltonian-connected* if such a path exists between any pair of distinct vertices. For notational convenience, we use $[n]$ to denote the set $\{1, 2, \ldots, n\}$.

The k-Ary n-Cube (Q_n^k) Topology

The k-ary n-cube (Q_n^k) is a foundational topology in parallel computing. Its structure is formally defined on a vertex set $V(Q_n^k) = \{0, 1, \ldots, k-1\}^n$, where each vertex is an n-digit vector in base k. Two vertices are connected by an edge if and only if their vector representations differ in exactly one coordinate i, where their values are adjacent modulo k (i.e., $u_i \equiv v_i \pm 1 \pmod{k}$). This definition yields a $2n$-regular graph with k^n vertices, a node and edge connectivity of $2n$, and a diameter of $n\lfloor k/2 \rfloor$ (Dally 2002; Ghozati and Wasserman 1999; Mao and Nicol 2003; Sarbazi-Azad et al. 2004).

A key property of the Q_n^k for fault tolerance analysis is its recursive structure. By partitioning vertices based on their value in a chosen dimension i, the graph is divided into k disjoint subgraphs, $Q[0], \ldots, Q[k-1]$. Each subgraph $Q[l]$ is induced by vertices whose i-th coordinate is l and is isomorphic to a lower-dimensional k-ary $(n-1)$-cube, Q_{n-1}^k. This structural decomposition provides the mathematical framework for our RBF model. To formalize this, we denote the unique neighbor of a vertex $u \in V(Q[l])$ in an adjacent subgraph $Q[l']$ as $n^{l'}(u)$, where $l' \equiv l \pm 1 \pmod{k}$. Consequently, $F_i[l, l+1]$ is the set of faulty edges in $F \cap E_i$ connecting regions $Q[l]$ and $Q[l+1]$. Figure 1 illustrates this topology.



(a) A 3-ary 3-cube (Q_3^3).

(b) RBF model in a 3-ary 3-cube.

Figure 1: Illustration of the RBF model in a 3-ary 3-cube (Q_3^3). (a) The standard topology. (b) An example demonstrating two distinct fault clusters (F_1, F_2) constrained by a minimum separation distance (S_d), which are core concepts of the RBF model.

The Region-Based Fault (RBF) Model

To model the clustered nature of real-world failures, we introduce the Region-Based Fault (RBF) model. RBF formally captures the spatial and topological characteristics of fault patterns, a paradigm shift from traditional approaches.

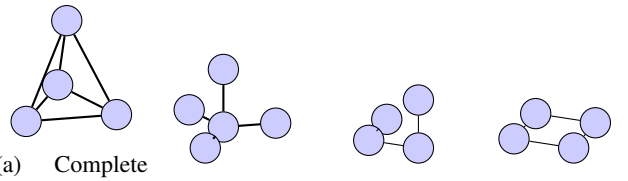
Core Concepts: Fault Clusters

The fundamental unit of failure in the RBF model is the fault cluster.

A **fault cluster** C is defined as a connected component of the fault-induced graph $G_F = (V_F, F)$, where $V_F = \{w \in V(Q_n^k) \mid \exists (u, v) \in F$ such that $w = u$ or $w = v\}$ is the set of all nodes incident to faulty edges, and F is the set of faulty edges. Each cluster represents a single, physically correlated failure event.

The model is defined by three key parameters that characterize these clusters:

- **Cluster Separation Distance:** For two distinct fault clusters C_i and C_j, their separation is defined as $d(C_i, C_j) = \min_{u \in V(C_i), v \in V(C_j)} d_H(u, v)$, where $d_H(u, v)$ is the Hamming distance between nodes u and v. This enforces that distinct failure events are spatially isolated.



(a) Complete (K_4) (b) Star (S_4) (c) Path (P_4) (d) Cycle (C_4)

Figure 2: Examples of allowed fault cluster shapes, assuming the total number of faulty nodes per cluster is bounded.

- **Cluster Size:** The size of a cluster is the number of faulty edges it contains. Bounding this parameter limits the impact of a single failure event.
- **Cluster Shape:** The shape is determined by the topology of the cluster's induced subgraph. We define a set of allowed shapes, $\mathcal{S}$, to model common failure patterns, such as a complete graph (e.g., a failed router), a star (a failed switch port), a path/tree, or a cycle, as illustrated in Figure 2.

Formal Definition of the RBF Model

Building upon these concepts, we provide the formal definition of the RBF model.

Definition 1 (Region-Based Fault (RBF) Model)**.** *Let $F \subseteq E(Q_n^k)$ be a set of faulty edges. Let $G_F = (V_F, F)$ be the fault-induced graph, where V_F is the set of nodes incident to edges in F. A fault cluster decomposition of F is a partition $\mathcal{C} = \{C_1, C_2, \ldots, C_m\}$, where each $C_i = (V(C_i), E(C_i))$ is a connected component of G_F.*

*The set F is a valid **RBF fault set** characterized by the parameter tuple $(k_{\max}, s_{\max}, d_{sep}, \mathcal{S})$ if its decomposition $\mathcal{C}$ satisfies the following four constraints:*

1. ***Cluster Count Bound:** The total number of clusters is bounded, i.e., $m = |\mathcal{C}| \leq k_{\max}$.*
2. ***Cluster Size Bound:** The size of any individual cluster, measured by its number of faulty edges, is limited, i.e., $\forall C_i \in \mathcal{C}, |E(C_i)| \leq s_{\max}$.*
3. ***Cluster Separation Bound:** Any two distinct clusters are spatially isolated by a minimum Hamming distance, i.e., $\forall C_i, C_j \in \mathcal{C}$ with $i \neq j$, $d(C_i, C_j) = \min_{u \in V(C_i), v \in V(C_j)} d_H(u, v) \geq d_{sep}$.*
4. ***Cluster Shape Constraint:** The topology of each cluster's induced subgraph belongs to a predefined set of allowed shapes, i.e., $\forall C_i \in \mathcal{C}, shape(C_i) \in \mathcal{S}$.*

The specific feasibility constraint on this parameter tuple that guarantees Hamiltonian connectivity is established in Theorem 1.

Properties and Advantages of the RBF Model

The RBF model's power stems from its core structural properties, which fundamentally bridge the gap between theoretical analysis and the physical reality of system failures. By representing faults as connected clusters with a bounded size ($|E_i| \leq s_{\max}$) and enforcing a minimum spatial separation between them ($d(C_i, C_j) \geq d_{\text{sep}}$), the model ensures that the influence regions of distinct failure events are localized and disjoint. This structural constraint directly translates into a significant advantage: enhanced analytical precision. Unlike traditional models that assume uniform fault distributions, RBF can accurately account for the clustered nature of real-world events like rack-level outages, leading to more realistic resilience estimates. Furthermore, this explicit spatial awareness creates new opportunities for algorithmic optimization, enabling the design of strategies that can isolate and efficiently route around fault zones—a capability that models lacking spatial context cannot exploit. The parameterized nature of the framework, defined by the tuple $(k_{\max}, s_{\max}, d_{\text{sep}}, \mathcal{S})$, adds practical utility, offering system architects the flexibility to tailor the model to specific hardware architectures and environmental conditions. For detailed guidance on selecting these parameters in practice, see Appendix C.

Fault-Tolerant Hamiltonian Path Embedding

Theoretical Foundation

In this section, we establish the theoretical foundation for our algorithm and prove its correctness under a sufficient set of conditions.

Basic Definitions and Properties

Definition 2 (Fault Cluster Properties)**.** *Let C_i be a fault cluster with edge set E_i and let $V(C_i)$ denote the set of nodes incident to edges in E_i. The cluster has the following properties:*

- *Size: $|E_i| \leq s_{\max}$*
- *Shape: shape$(C_i) \in \mathcal{S}$*
- *Affected nodes: $|V(C_i)| \leq 2s_{\max}$*

Lemma 1 (Bounding Cluster Diameter)**.** *Let C_i be a fault cluster of size $|E_i| \leq s_{\max}$. The Hamming diameter of its affected node set, diam$_H(V(C_i)) = \max_{u,v \in V(C_i)} d_H(u, v)$, is bounded by $s_{\max}$.*

Proof. For any two nodes $u, v \in V(C_i)$, a simple path of edges from E_i connects them, as C_i is connected. Let the length of the shortest such path be l. By the triangle inequality, the Hamming distance $d_H(u, v)$ is less than or equal to the sum of the Hamming distances of the edges along this path. Since each edge in Q_n^k has a Hamming distance of 1, we have $d_H(u, v) \leq l$.

The maximum length of a shortest path between any two nodes in the cluster graph C_i is its diameter. From the case analysis on allowed shapes in $\mathcal{S}$, the maximum number of nodes in a cluster is at most $s_{\max} + 1$ (for a path/tree shape), and the diameter of the cluster graph is at most $s_{\max}$. Thus, the maximum shortest path length l is at most $s_{\max}$. Therefore, $\text{diam}_H(V(C_i)) = \max_{u,v \in V(C_i)} d_H(u, v) \leq s_{\max}$. □

Core Theoretical Results Our central theoretical result establishes the sufficient conditions for Hamiltonicity under the RBF model.

Lemma 2 (Cross-Layer Edge Availability)**.** *Let F be a fault set satisfying RBF conditions $(k_{\max}, s_{\max}, d_{sep}, \mathcal{S})$ in Q_n^k. Let the network be decomposed along dimension d^*. The number of faulty edges $|F_{j,j+1}|$ between any adjacent subcubes $Q[j]$ and $Q[j+1]$ is bounded by:*

$$|F_{j,j+1}| \leq s_{\max} \cdot \min\left(k_{\max}, K_{bound}(n, k, d_{sep})\right)$$

where $K_{bound}(n, k, d_{sep}) = \frac{k^{n-1}}{\sum_{i=0}^{\lfloor (d_{sep}-1)/2 \rfloor} \binom{n-1}{i}(k-1)^i}$ is the Hamming bound on the number of clusters that can simultaneously damage the interface.

Proof. **Step 1: Bounding the number of "damaging" clusters.** Let $\mathcal{C}_{j,j+1} \subseteq \mathcal{C}$ be the set of fault clusters that damage the interface between $Q[j]$ and $Q[j+1]$. For any cluster $C_i \in \mathcal{C}_{j,j+1}$, its node set $V(C_i)$ must contain at least one node in $Q[j]$ and one node in $Q[j+1]$.

For each such cluster $C_i \in \mathcal{C}_{j,j+1}$, we can select a representative node $u_i \in V(C_i) \cap V(Q[j])$. Now, consider any two distinct clusters $C_i, C_l \in \mathcal{C}_{j,j+1}$ with their representative nodes $u_i, u_l \in V(Q[j])$.

By the RBF condition, the Hamming distance between any node in C_i and any node in C_l is at least d_{sep}. Thus, $d_H(u_i, u_l) \geq d_{\text{sep}}$. Since both u_i and u_l are in the subcube $Q[j]$, their d^*-th coordinates are identical. This means their Hamming distance is entirely determined by the other $n-1$ coordinates.

Step 2: Applying a packing argument in the subcube. We now have a set of nodes $\{u_i\}_{i \in \mathcal{I}}$ within the subcube $Q[j]$ (which is isomorphic to Q_{n-1}^k) such that the pairwise Hamming distance is at least d_{sep}. The size of this set, $|\mathcal{C}_{j,j+1}|$, is the number of damaging clusters.

We can bound this size using a standard sphere-packing argument (the Hamming bound). The ball $B(u, r)$ of radius r around a node u in Q_{n-1}^k contains $\sum_{i=0}^{r} \binom{n-1}{i}(k-1)^i$ nodes. If we place disjoint balls of radius $r = \lfloor (d_{\text{sep}}-1)/2 \rfloor$ around each representative node u_i, these balls must all fit within the total volume of the subcube Q_{n-1}^k, which has k^{n-1} nodes.

Therefore, the number of damaging clusters is bounded by:

$$|\mathcal{C}_{j,j+1}| \leq \frac{|V(Q_{n-1}^k)|}{|B(u,r)|} = \frac{k^{n-1}}{\sum_{i=0}^{\lfloor (d_{\text{sep}}-1)/2 \rfloor} \binom{n-1}{i}(k-1)^i} =: K_{bound}$$

This bound is derived from established combinatorial principles and is therefore rigorous. For any non-trivial $d_{\text{sep}} \geq 3$, the denominator grows, making K_{bound} a powerful constraint.

Step 3: Calculating the total faulty edges. Each of the $|\mathcal{C}_{j,j+1}|$ damaging clusters can contribute at most $s_{\max}$ faulty edges in total. Thus, the total number of faulty edges on the interface is bounded by the product of the maximum number of damaging clusters and the maximum size of each cluster:

$$|F_{j,j+1}| \leq |\mathcal{C}_{j,j+1}| \cdot s_{\max} \leq K_{bound}(n, k, d_{\text{sep}}) \cdot s_{\max}$$

Step 4: Finalizing the bound with a dual constraint. The sphere-packing argument in Step 2 provides a powerful *local geometric bound*, K_{bound}, on the number of clusters that can physically affect the interface. However, this must be reconciled with the *global model bound*: the number of damaging clusters, $|\mathcal{C}_{j,j+1}|$, obviously cannot exceed the total number of clusters, $k_{\max}$, permitted in the entire network by the RBF model's definition. To form the tightest possible (most stringent) mathematical upper bound, we must enforce both of these independent constraints simultaneously. Therefore, the effective maximum number of damaging clusters is the minimum of the global limit and the local geometric limit. Multiplying by the maximum size of each cluster, $s_{\max}$, yields the final expression:

$$|F_{j,j+1}| \leq s_{\max} \cdot \min\left(k_{\max}, K_{bound}(n, k, d_{\text{sep}})\right)$$

This demonstrates mathematically how separation distance limits the "conspiracy" of clusters against a single interface. It is important to note that K_{bound} represents a worst-case packing limit; as shown experimentally, typical cluster placements result in far fewer damaging clusters. □

Lemma 3 (Inductive Property Preservation)**.** *Let F be a fault set satisfying RBF conditions $(k_{\max}, s_{\max}, d_{sep}, \mathcal{S})$ in Q_n^k. When decomposed along dimension d^*, the induced fault set F_j within any subcube $Q[j]$ satisfies a set of RBF-like conditions $(k'_{\max}, s'_{\max}, d'_{sep}, \mathcal{S}')$. Specifically, $s'_{\max} \leq s_{\max}$ and $d'_{sep} \geq d_{sep}$. The number of clusters $k'_{\max}$ is bounded by $k'_{\max} \leq k_{\max} \cdot s_{\max}$.*

Proof. Let $\mathcal{C}$ be the cluster decomposition of F in Q_n^k. The induced fault set F_j in a subcube $Q[j]$ consists of all edges from F that are internal to $Q[j]$.

- **Size ($s'_{\max}$):** Any new cluster in $Q[j]$ is a subgraph of an original cluster from $\mathcal{C}$. Therefore, its size cannot exceed the original maximum size, i.e., $s'_{\max} \leq s_{\max}$.
- **Separation (d'_{sep}):** The Hamming distance between any two nodes within the subcube $Q[j]$ is calculated over $n-1$ dimensions. This distance is never less than their distance in the full n-dimensional space. Thus, the minimum separation between any two new clusters in $Q[j]$ is at least d_{sep}, so $d'_{\text{sep}} \geq d_{\text{sep}}$.
- **Number of clusters ($k'_{\max}$):** A single cluster $C_i \in \mathcal{C}$ that spans multiple subcubes may be partitioned into several disconnected components within a single subcube $Q[j]$. A cluster with $|E_i| \leq s_{\max}$ edges has at most $s_{\max} + 1$ nodes (if it's a tree/path). The number of components it can split into within $Q[j]$ is bounded by its number of edges, $s_{\max}$. In the worst case, all $k_{\max}$ clusters are split this way, leading to a new maximum cluster count of $k'_{\max} \leq k_{\max} \cdot s_{\max}$.

The shapes of the new cluster components, $\mathcal{S}'$, will be subgraphs (typically forests) of the original shapes in $\mathcal{S}$. The induced fault set thus satisfies a well-defined, albeit potentially weaker, set of RBF-like conditions. □

Theorem 1 (RBF Hamiltonian Connectivity)**.** *For odd $k \geq 3$ and $n \geq 2$, if a fault set $F \subseteq E(Q_n^k)$ satisfies the RBF conditions $(k_{\max}, s_{\max}, d_{sep}, \mathcal{S})$ and the constraint:*

$$\min\left(k_{\max}, K_{bound}(n, k, d_{sep})\right) \cdot s_{\max} < \frac{k^{n-1}}{4}$$

where $K_{bound}(n, k, d_{sep})$ is defined as in Lemma 2, then $Q_n^k - F$ remains Hamiltonian-connected.

Proof. The proof proceeds by induction on n.

Base case ($n = 2$): For Q_2^k (a torus), the number of cross-layer edges is k. The condition becomes $\min(k_{\max}, K_{bound}) \cdot s_{\max} < k/4$. This ensures that the fault density is sufficiently low for known algorithms to construct Hamiltonian paths in faulty 2D tori.

Inductive step: Assume the theorem holds for dimensions up to $n-1$. Consider Q_n^k with an RBF fault set F satisfying the theorem's condition.

1. Select an optimal decomposition dimension d^* that maximizes fault cluster separation. 2. Decompose Q_n^k along d^* into k subcubes $Q[0], Q[1], \ldots, Q[k-1]$, each isomorphic to Q_{n-1}^k. The total number of edges between any adjacent subcubes $Q[j]$ and $Q[j+1]$ is k^{n-1}. 3. By Lemma 3, the induced fault set F_j within each subcube $Q[j]$ satisfies a set of RBF-like conditions. For the inductive hypothesis to apply, we must ensure these induced conditions are sufficiently strong. As $s'_{\max} \leq s_{\max}$ and $d'_{\text{sep}} \geq d_{\text{sep}}$, the critical condition on cross-layer faults for the subproblem (of dimension $n-1$) will also hold. 4. By the inductive hypothesis, each subcube $Q[j]$ that is not overly damaged is Hamiltonian-connected. 5. By the newly proven Lemma 2, the number of faulty cross-layer edges $|F_{j,j+1}|$ is bounded by $F_{upper} = \min(k_{\max}, K_{bound}(n, k, d_{\text{sep}})) \cdot s_{\max}$. 6. The condition of the theorem, $F_{upper} < k^{n-1}/4$, guarantees that the number of healthy cross-layer edges between $Q[j]$ and $Q[j+1]$ is at least $k^{n-1} - F_{upper} > k^{n-1} - k^{n-1}/4 = (3/4)k^{n-1}$. This vast number of available edges ensures that the path stitching step in our recursive algorithm will succeed.

Therefore, by recursively finding Hamiltonian paths in the subcubes and stitching them together using the abundant healthy cross-layer edges, a global Hamiltonian path can be constructed between any two nodes. □

Algorithm Design and Correctness

This section presents our RBF-based Hamiltonian path construction algorithm and its proof of correctness.

Algorithm Overview Our algorithm employs a recursive divide-and-conquer approach specifically designed for the RBF model. The primary contribution consists of an adaptive dimension selection strategy that maximizes fault cluster separation. The overall procedure, shown in Algorithm 1, first validates the RBF conditions. The `SelectOptimalDimension` function (line 8) implements a crucial heuristic to find a decomposition that maximizes fault isolation. It iterates through all possible dimensions $d \in \{0, \ldots, n-1\}$. For each dimension d, it calculates a cost score, defined as the number of fault clusters that have edges in more than one subcube if the network were to be decomposed along dimension d. The function then returns the dimension d^* with the minimum cost:

$$d^* = \underset{d \in \{0,\ldots,n-1\}}{\operatorname{argmin}} \sum_{i=1}^{|\mathcal{C}|} \mathbb{I}(\text{spans\_dimension}(C_i, d))$$

Algorithm 1 RBF Hamiltonian Path Construction

Require: Q_n^k: k-ary n-cube network, F: RBF fault edge set, s, t: source and target nodes
Ensure: Hamiltonian path from s to t, or NULL if impossible

1: $\mathcal{C} \leftarrow$ AnalyzeFaultClusters(F)
2: **if** Theorem 1 condition is not met **then**
3: *// Proceed with best-effort, as condition is sufficient but not necessary*
4: **end if**
5: **if** $n = 2$ **then**
6: **return** HamiltonianPath2D(Q_2^k, F, s, t)
7: **end if**
8: $d^* \leftarrow$ SelectOptimalDimension($\mathcal{C}, n$)
9: $\{Q[0], \ldots, Q[k-1]\} \leftarrow$ DecomposeNetwork(Q_n^k, d^*)
10: SubPaths $\leftarrow$ []
11: **for** $i = 0$ to $k-1$ **do**
12: $F_i \leftarrow F \cap E(Q[i])$
13: **if** $i = s$.layer **then**
14: start_node $\leftarrow s$
15: **else**
16: start_node $\leftarrow$ SelectStartNode($Q[i]$, SubPaths$[i-1]$)
17: **end if**
18: **if** $i = t$.layer **then**
19: end_node $\leftarrow t$
20: **else**
21: end_node $\leftarrow$ SelectEndNode($Q[i], Q[i+1], d^*$)
22: **end if**
23: $P_i \leftarrow$ RBF_HamiltonianPath($Q[i], F_i$, start_node, end_node)
24: **if** $P_i =$ NULL **then**
25: **return** NULL
26: **end if**
27: SubPaths.append(P_i)
28: **end for**
29: **return** StitchPaths(SubPaths, d^*)

where $\mathbb{I}(\cdot)$ is the indicator function and spans_dimension(C_i, d) is true if cluster C_i contains nodes whose d-th coordinates differ. This adaptive selection is critical, as it allows the algorithm to proactively find a decomposition that minimizes inter-subcube disruption, often resulting in performance far exceeding the worst-case theoretical guarantees. For $n > 2$, it recursively decomposes the network along the selected optimal dimension, solves for Hamiltonian paths in the subcubes, and then stitches these paths together to form a global Hamiltonian path.

Algorithm Complexity Analysis

The time complexity of Algorithm 1 is $O(n \cdot k^n)$ and space complexity is $O(k^n)$. The algorithm achieves near-optimal performance with only a factor of n overhead compared to the theoretical lower bound, which comes from the adaptive dimension selection process that is essential for the improved fault tolerance.

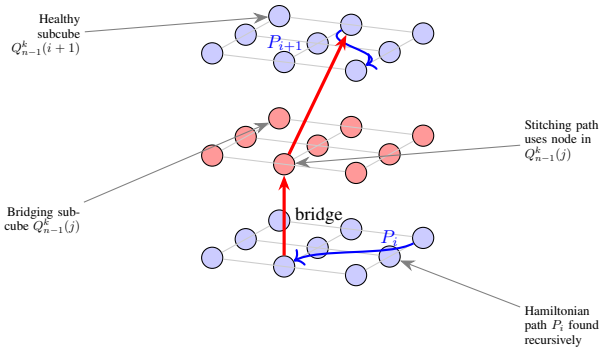


Figure 3: 3D illustration of the recursive path stitching mechanism.

Table 1: Fault tolerance comparison for RBF, PEF, and FT models.

Network Config.	RBF Tol.	PEF Tol.	FT Tol.
3-ary 5-cube (Q_5^3)	41	24	3
4-ary 4-cube (Q_4^4)	134	77	5
5-ary 5-cube (Q_5^5)	1,607	770	7
6-ary 4-cube (Q_4^6)	2,648	1,353	9
7-ary 4-cube (Q_4^7)	10,403	5,447	11
8-ary 3-cube (Q_3^8)	5,934	3,265	13
9-ary 3-cube (Q_3^9)	17,561	9,824	15
10-ary 3-cube (Q_3^{10})	52,009	29,505	17

Experimental Evaluation and Analysis

We now experimentally evaluate our algorithm's performance and validate our claims of enhanced fault tolerance.

Experimental Setup

The simulations were performed using a custom-built Python framework implementing the Q_n^k topology and our RBF-based algorithm. We compared our RBF model against the traditional Fault-Tolerant (FT) model and the Partitioned Edge Fault (PEF) model. The experiments covered network configurations with dimensions n from 3 to 10 and radix k from 3 to 10.

Fault Tolerance Benchmarking

Our experiments quantified the fault tolerance improvement of the RBF model. Table 1 shows that the RBF model consistently tolerates significantly more faults than competing approaches.

Structural and Spatial Sources of Resilience

Our experimental results consistently show that the RBF algorithm tolerates significantly more faults than predicted by prior models. To explain this performance gain, we introduce a correction factor model. This model serves as an analytical tool, not part of the formal proof, to decompose the sources of resilience. The effective fault tolerance is modeled as:

$$\Theta_{RBF} = k_{\max} \cdot s_{\max} \cdot \alpha_{struct} \cdot \alpha_{spatial}$$

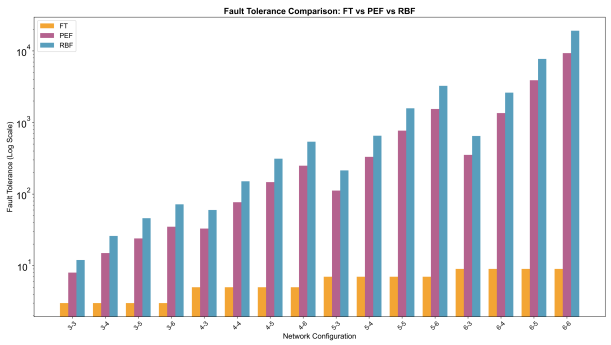


Figure 4: Our RBF model achieves orders-of-magnitude greater fault tolerance than prior art. The logarithmic scale on the Y-axis highlights that across all tested network configurations, our approach (blue) dramatically outperforms both the PEF (purple) and traditional FT (orange) models, with the performance gap widening as network scale increases.

Here, α_{struct} captures the structural effect, where our adaptive algorithm finds a decomposition that isolates faults more effectively than the worst case. Concurrently, $\alpha_{spatial}$ represents the spatial effect, where the d_{sep} constraint limits fault packing—a principle mathematically grounded in the Hamming bound of Lemma 2. Figure 5 visualizes the distinct contributions of these factors, together explaining the observed performance improvement.

Parameter Sensitivity Analysis Figure 6 shows how RBF parameters influence performance on a 6-ary 5-cube network. Fault tolerance scales positively with both k_{max} and s_{max}, shows positive correlation with d_{sep}, and decreases with higher spatial correlation factor ρ.

Theoretical Bounds Tightness and Robustness

Our algorithm demonstrates striking robustness, often succeeding even when the sufficient condition of Theorem 1 is not met. This resilience stems from three factors: the Hamming bound used in K_{bound} is highly conservative (our experiments show the measured fault load is only 21.2% of this bound on average); the proof's worst-case fault placements are rare in practice; and most importantly, our adaptive dimension selection strategy actively finds a decomposition that avoids the worst case. This combined effect is quantified in Figure 7, which shows the algorithm handles on average 46% more faults than the formal guarantee, corresponding to a tightness ratio of 1.459.

Geometric Analysis and Boundary Behavior

The RBF model's ability to differentiate between fault cluster shapes provides significant advantages. Figure 9(a) compares properties of Complete Graph and Star Graph clusters, showing that Complete clusters exhibit higher density (0.751 vs 0.429) and smaller diameter, enabling more effective isolation and routing strategies.

Figure 9(b) explores the algorithm's behavior around the theoretical boundary. The performance ratio remains stable and even increases as we approach and slightly exceed the

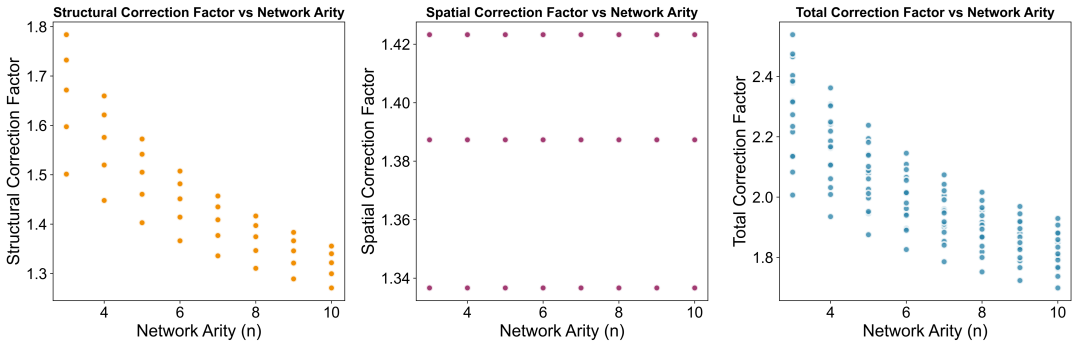


Figure 5: Analysis of structural, spatial, and total correction factors showing distinct contributions of network topology and fault distribution.

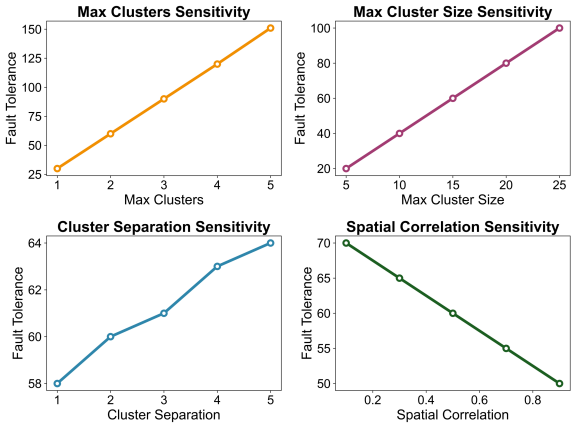


Figure 6: Parameter sensitivity analysis showing impact on fault tolerance.

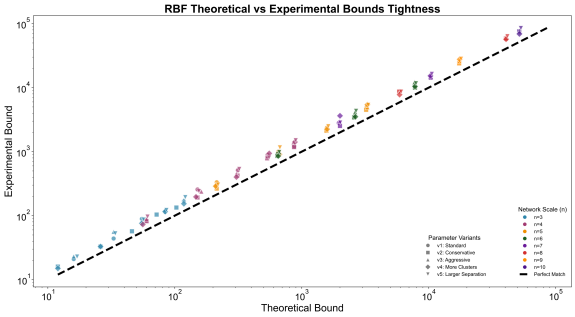


Figure 7: Tightness analysis demonstrating model accuracy and conservative guarantees.

boundary. This indicates that the condition in Theorem 1 is not a "hard cliff" where performance abruptly collapses, but rather a "soft boundary" marking the transition from guaranteed success to high-probability success. This graceful degradation is a highly desirable property for real-world systems.

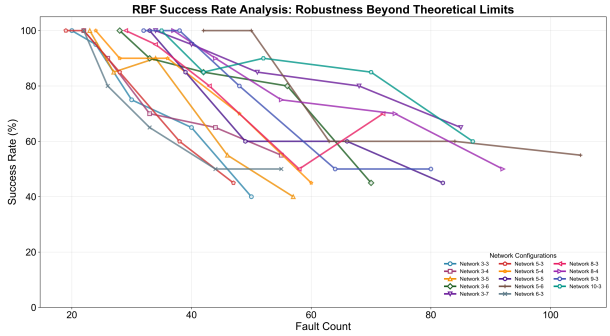
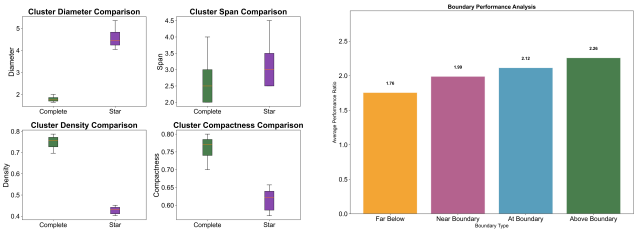


Figure 8: Success rate analysis showing graceful degradation beyond theoretical limits.



(a) Geometric properties of fault clusters.

(b) Boundary and connectivity analysis.

Figure 9: Analysis of fault cluster geometry and performance near the theoretical boundary.

Experimental Validation of Theoretical Lemmas

To bridge the gap between our theoretical framework and its practical implications, we conducted a direct experimental validation of our core result, Lemma 2. We generated 340 distinct RBF fault configurations across a wide range of network sizes (from Q_3^3 to Q_7^4), varying parameters such as the separation distance d_{sep}. For each configuration, we calculated the theoretical upper bound on faulty cross-layer edges, F_{upper}, and measured the actual maximum number of such edges, $F_{\text{measured\_max}}$, across all possible dimensional decompositions.

The results, plotted in Figure 10, decisively validate our

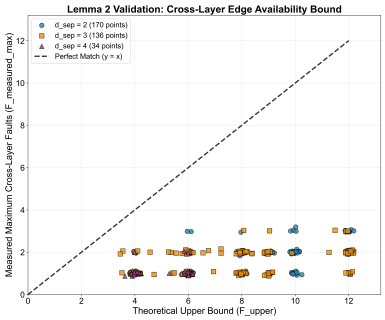


Figure 10: Validation of Lemma 2, showing the measured maximum faulty edges (Y-axis) are always bounded by the theoretical limit (X-axis). The gap below the identity line highlights the bound's conservative nature, explaining the algorithm's practical robustness.

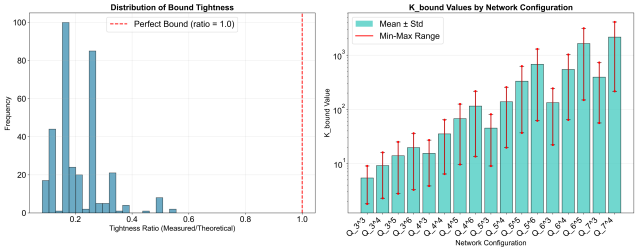


Figure 11: Analysis of Lemma 2 validation. (a) The tightness ratio averages only 0.212, confirming the bound's conservative nature. (b) The K_{bound} (log scale) demonstrates its effectiveness by decreasing as network size and fault separation increase.

theory. As predicted, all 340 data points fall on or below the $y = x$ line, empirically confirming that $|F_{\text{measured\_max}}| \leq F_{upper}$ in every tested case. This provides powerful empirical evidence for the correctness of our combinatorial argument. Furthermore, the analysis reveals that the Hamming bound, while rigorously proven, provides a conservative estimate. The average tightness ratio (the ratio of measured faults to the theoretical bound) was only 0.212, as detailed in Figure 11. This built-in safety margin is a key reason for the algorithm's strong robustness.

Discussion

Spatial Arrangement as a Factor in System Resilience

Our work suggests that in resilient system design, the spatial arrangement of faults can be a more critical factor than their aggregate number. We translate this concept into a quantitative method. The `K_bound` formula (from Lemma 2) offers a way for system architects to perform trade-off analysis during the design phase. This can help in justifying investments in physical fault isolation (Narayanan, Shibley, and Pundir 2020; Microsoft 2024; Gill, Jain, and Nagappan 2011) and allows architects to answer questions such as, "To maintain network connectivity against the failure of any three adjacent racks, what is the minimum required physical separation `dsep`?"

Connecting Spatial Awareness to Deployment Resilience

Our RBF framework addresses a key challenge in large-scale AI deployments: the fragility of the underlying physical infrastructure. It provides a method to mitigate training failures caused by localized physical events, such as a full rack failure, by working to preserve network properties like Hamiltonian connectivity. This suggests that strategic physical design, for example, increasing the separation distance `dsep` between failure domains, may offer a cost-effective path to resilience compared to solely investing in more expensive hardware. Furthermore, RBF could be integrated into topology-aware schedulers to enforce anti-affinity rules, improving an application's fault tolerance by preventing redundant services from being co-located in the same physical fault region.

Limitations and Future Work

Our analysis presents several avenues for future research. The current RBF model is based on a predefined set of fault cluster shapes, $\mathcal{S}$. Its theoretical bound, `K_bound`, is derived from the Hamming bound (Hamming 1950), which can be conservative. Future work could use graph-based machine learning to dynamically identify fault regions from system telemetry, removing the dependency on predefined shapes. Additionally, investigating tighter combinatorial bounds (e.g., Plotkin, Johnson, or Gilbert-Varshamov) could refine our sufficient condition. This would potentially widen the range of provably fault-tolerant configurations and strengthen the engineering guarantees our framework provides.

Conclusion

This paper addressed the fragility of large-scale computing infrastructure, particularly in the context of physically clustered failures that are common in AI deployments. We introduced the Region-Based Fault (RBF) framework, a model that treats physical layout as a key design constraint, moving beyond traditional fault models that often neglect spatial information. Within this framework, we formalized the principle that the geometric dispersion of faults is a more significant determinant of network resilience than their aggregate count. Our constructive algorithm and experimental results validate this principle, demonstrating robust performance against realistic failure scenarios. The connectivity guarantees provided by our method can support more reliable deadlock-free routing and load balancing (Dally and Seitz 1987; Dally and Towles 2004; Gaughan et al. 2002), helping to maintain high performance for AI workloads during hardware failures. A potential direction for future work is to use graph neural networks for automated RBF parameterization from system telemetry (Huoh et al. 2022; Pang et al. 2023; Le et al. 2021). In summary, our work provides a formal basis and practical guidance for designing the next generation of resilient AI infrastructure.

Acknowledgments

This work was supported by the Xinjiang Uygur Autonomous Region University Student Innovation and Entrepreneurship Training Program (Project No. S202410755159).

Data Availability

The code for this study is available at https://github.com/wyqmath/Hamiltonian_Path.

References

Abd-El-Barr, M.; and Gebali, F. 2014. Reliability analysis and fault tolerance for hypercube multi-computer networks. *Information Sciences*, 276: 295–318.

Ali, R.; Abbas, A. M.; and Khan, I. A. 2004. Feasibility of a fault tolerant routing algorithm for hypercube multiprocessors. In *Proceedings of the IEEE INDICON 2004. First India Annual Conference, 2004.*, 419–422. IEEE.

Ashir, Y. A.; and Stewart, I. A. 2002. Fault-tolerant embeddings of Hamiltonian circuits in k-ary n-cubes. *SIAM Journal on Discrete Mathematics*, 15(3): 317–328.

Bondy, J. A.; and Murty, U. S. R. 1976. *Graph theory with applications*, volume 290. Macmillan.

Chang, Y.; and Bhuyan, L. N. 2002. Subcube fault tolerance in hypercube multiprocessors. *IEEE transactions on computers*, 44(9): 1108–1120.

Chen, K. C. J.; Chao, C. H.; and Wu, A. Y. A. 2015. Thermal-aware 3D network-on-chip (3D NoC) designs: Routing algorithms and thermal managements. *IEEE circuits and Systems Magazine*, 15(4): 45–69.

Dally, W. J. 2002. Performance analysis of k-ary n-cube interconnection networks. *IEEE transactions on Computers*, 39(6): 775–785.

Dally, W. J.; and Seitz, C. L. 1987. Deadlock-free message routing in multiprocessor interconnection networks. *IEEE Transactions on computers*, C-36(5): 547–553.

Dally, W. J.; and Towles, B. P. 2004. *Principles and practices of interconnection networks*. Elsevier.

Dean, J.; Corrado, G.; Monga, R.; Chen, K.; Devin, M.; Mao, M.; ...; and Ng, A. 2012. Large scale distributed deep networks. In *Advances in neural information processing systems*, volume 25.

Dong, H.; Wang, H.; Lv, M.; and Fan, W. 2025. The Hamiltonian Property of the Data Center Network DPCell. *IEEE Transactions on Network Science and Engineering*.

Feero, B. S.; and Pande, P. P. 2008. Networks-on-chip in a three-dimensional environment: A performance evaluation. *IEEE Transactions on computers*, 58(1): 32–45.

Gaughan, P. T.; Dao, B. V.; Yalamanchili, S.; and Schimmel, D. E. 2002. Distributed, deadlock-free routing in faulty, pipelined, direct interconnection networks. *IEEE Transactions on Computers*, 45(6): 651–665.

Ghozati, S. A.; and Wasserman, H. C. 1999. The k-ary n-cube network: modeling, topological properties and routing strategies. *Computers & electrical engineering*, 25(3): 155–168.

Gill, P.; Jain, N.; and Nagappan, N. 2011. Understanding network failures in data centers: measurement, analysis, and implications. In *Proceedings of the ACM SIGCOMM 2011 Conference*, 350–361.

Gould, R. 2012. *Graph theory*. Courier Corporation.

Gross, J. L.; Yellen, J.; and Anderson, M. 2018. *Graph theory and its applications*. Chapman and Hall/CRC.

Gu, R.; Gu, X.; Shi, Y.; and Wang, H. 2022. ℓ-Connectivity and ℓ-edge-connectivity of random graphs. *Journal of Graph Theory*, 101(1): 5–28.

Gu, R.; Shi, Y.; and Fan, N. 2017. Mixed connectivity of random graphs. In *International Conference on Combinatorial Optimization and Applications*, 133–140. Cham: Springer International Publishing.

Hacker, T. J.; Romero, F.; and Carothers, C. D. 2009. An analysis of clustered failures on large supercomputing systems. *Journal of Parallel and Distributed Computing*, 69(7): 652–665.

Hamming, R. W. 1950. Error detecting and error correcting codes. *The Bell system technical journal*, 29(2): 147–160.

Huoh, T. L.; Luo, Y.; Li, P.; and Zhang, T. 2022. Flow-based encrypted network traffic classification with graph neural networks. *IEEE Transactions on Network and Service Management*, 20(2): 1224–1237.

Jouppi, N.; Kurian, G.; Li, S.; Ma, P.; Nagarajan, R.; Nai, L.; ...; and Patterson, D. A. 2023. TPU v4: An optically reconfigurable supercomputer for machine learning with hardware support for embeddings. In *Proceedings of the 50th Annual International Symposium on Computer Architecture*, 1–14.

Jouppi, N. P.; Young, C.; Patil, N.; Patterson, D.; Agrawal, G.; Bajwa, R.; ...; and Yoon, D. H. 2017. In-datacenter performance analysis of a Tensor Processing Unit. In *Proceedings of the 44th annual international symposium on computer architecture*, 1–12.

Le, T. T.; Le Nguyen, P.; Binh, H. T. T.; Akerkar, R.; and Ji, Y. 2021. GCRINT: network traffic imputation using graph convolutional recurrent neural network. In *ICC 2021-IEEE International Conference on Communications*, 1–6. IEEE.

Liu, X.; Meng, J.; and Sabir, E. 2023. Component connectivity of the data center network DCell. *Applied Mathematics and Computation*, 444: 127822.

Lv, M.; Li, Y.; Gao, H.; Sun, B.; Huang, K.; Yang, C.; and Gui, W. 2025. A hierarchical stochastic network approach for fault diagnosis of complex industrial processes. *IEEE/CAA Journal of Automatica Sinica*.

Lv, Y.; Lin, C. K.; Fan, J.; and Jia, X. 2018. Hamiltonian cycle and path embeddings in 3-ary n-cubes based on K1, 3-structure faults. *Journal of Parallel and Distributed Computing*, 120: 148–158.

Mao, W.; and Nicol, D. M. 2003. On k-ary n-cubes: theory and applications. *Discrete applied mathematics*, 129(1): 171–193.

Microsoft. 2024. Datacenter architecture and infrastructure. Microsoft Learn. Available at: https://learn.microsoft.

com/en-us/compliance/assurance/assurance-datacenter-architecture-infrastructure. Accessed: 2025-08-01.

Narayanan, A.; Shibley, E.; and Pundir, M. 2020. Fault tolerance through optimal workload placement. https://engineering.fb.com/2020/09/08/data-center-engineering/fault-tolerance-through-optimal-workload-placement/. Accessed: 2025-08-01.

NVIDIA. 2025. DGX SuperPOD Architecture. https://docs.nvidia.com/dgx-superpod/reference-architecture-scalable-infrastructure-h100/latest/dgx-superpod-architecture.html. Accessed: 2025-08-01.

Pang, B.; Fu, Y.; Ren, S.; and Jia, Y. 2023. High-performance network traffic classification based on graph neural network. In *2023 IEEE 6th Information Technology, Networking, Electronic and Automation Control Conference (ITNEC)*, volume 6, 800–804. IEEE.

Pinheiro, E.; Weber, W. D.; and Barroso, L. A. 2007. Failure Trends in a Large Disk Drive Population. In *FAST*, volume 7, 17–23.

Qiao, H.; and Zhang, W. 2024. On the Spanning Cyclability of k-ary n-cube Networks. *Symmetry*, 16(8): 1063.

Sabir, E.; Fan, J.; Meng, J.; and Cheng, B. 2023. Structure Fault-Tolerant Hamiltonian Cycle and Path Embeddings in Bipartite k-Ary n-Cube Networks. *IEEE Transactions on Reliability*, 73(1): 257–269.

Sabir, E.; and Lin, C. K. 2025. Structure Fault Tolerance of Fully Connected Cubic Networks. *Mathematics*, 13(9): 1532.

Sarbazi-Azad, H.; Ould-Khaoua, M.; and Mackenzie, L. M. 2001. An accurate analytical model of adaptive wormhole routing in k-ary n-cubes interconnection networks. *Performance Evaluation*, 43(2-3): 165–179.

Sarbazi-Azad, H.; Ould-Khaoua, M.; Mackenzie, L. M.; and Akl, S. G. 2004. On the combinatorial properties of k-ary n-cubes. *Journal of Interconnection Networks*, 5(01): 79–91.

Shinde, A.; and Borse, A. M. 2021. Disjoint cycles through prescribed vertices in multidimensional tori. *Journal of the Ramanujan Mathematical Society*, 36(4): 283–290.

Sun, X.; Fan, J.; Sabir, E.; Cheng, B.; and Yu, J. 2023. Reliability of augmented k-ary n-cubes under the extra connectivity condition: X. Sun et al. *The Journal of Supercomputing*, 79(12): 13641–13669.

Tutte, W. T. 2001. *Graph theory*, volume 21. Cambridge university press.

Vishwanath, K. V.; and Nagappan, N. 2010. Characterizing cloud computing hardware reliability. In *Proceedings of the 1st ACM symposium on Cloud computing*, 193–204.

Xia, Y.; Wang, C.; Shen, H. L.; and Song, H. 2020. Cascading failures in spatial complex networks. *Physica A: Statistical Mechanics and its Applications*, 559: 125071.

Yang, M. C.; Tan, J. J.; and Hsu, L. H. 2007. Hamiltonian circuit and linear array embeddings in faulty k-ary n-cubes. *Journal of Parallel and Distributed Computing*, 67(4): 362–368.

Zhang, H.; and Meng, J. 2021. Faulty diagnosability and g-extra connectivity of DQcube. *International Journal of Parallel, Emergent and Distributed Systems*, 36(3): 189–198.

Zhang, P.; and Chartrand, G. 2006. *Introduction to graph theory*, volume 2. Tata McGraw-Hill.

Zhuang, H.; Li, X. Y.; Chang, J. M.; Lin, C. K.; and Liu, X. 2023a. Embedding Hamiltonian Paths in k-Ary n-Cubes With Exponentially-Many Faulty Edges. *IEEE Transactions on Computers*, 72(11): 3245–3258.

Zhuang, H.; Li, X. Y.; Chang, J. M.; and Liu, X. 2024. Paired 2-disjoint path covers of k-ary n-cubes under the partitioned edge fault model. *Journal of Parallel and Distributed Computing*, 190: 104887.

Zhuang, H.; Li, X. Y.; Chang, J. M.; and Wang, D. 2023b. An efficient algorithm for hamiltonian path embedding of k-ary n-cubes under the partitioned edge fault model. *IEEE Transactions on Parallel and Distributed Systems*, 34(6): 1802–1815.

Appendix

A Detailed Review of Fault Tolerance Models

This appendix provides a more comprehensive review of the fault tolerance models discussed in the main text, offering greater technical detail on their assumptions, theoretical bounds, and limitations. This detailed analysis serves to further motivate the development of the Region-Based Fault (RBF) model.

The Traditional Fault-Tolerant (FT) Model

The most established approach to fault tolerance is the Traditional Fault-Tolerant (FT) model, which operates under a stringent worst-case assumption: faults are independent and can be arbitrarily distributed throughout the network. Research within this paradigm focuses on determining the maximum number of faults, denoted by cardinality $|F|$, that a network can withstand while preserving a desired property (e.g., Hamiltonian connectivity), irrespective of the faults' locations.

A landmark study for the Q_n^k topology by Yang, Tan, and Hsu (2007) demonstrated that for odd $k \geq 3$, the network remains Hamiltonian-connected as long as the number of faulty edges does not exceed $2n - 3$. While this result is mathematically optimal under the model's assumptions, its practical utility is limited. The FT model's primary drawback is its lack of realism, as it permits pathological fault distributions—such as all $2n - 3$ faults being incident to a single node—that are highly improbable in practice. This leads to overly pessimistic and often impractical assessments of a network's true resilience. This limitation has been a consistent theme in analyses of hypercubes, star graphs, and various other topologies under the FT model (Abd-El-Barr and Gebali 2014; Chang and Bhuyan 2002; Ashir and Stewart 2002; Lv et al. 2018; Sun et al. 2023; Shinde and Borse 2021; Lv et al. 2025).

The Partitioned Edge Fault (PEF) Model

To overcome the pessimism of the FT model, researchers developed Conditional Fault Models. These models impose more realistic constraints on fault distributions, typically based on the principle that a large number of faults are unlikely to conspire to isolate a single node or a small region. A significant evolution in this area is the Partitioned Edge Fault (PEF) model, which acknowledges that faults may not be uniformly distributed across a network's logical structure (Zhuang et al. 2023b, 2024, 2023a).

Motivated by high-performance architectures like 3D Networks-on-Chip (NoCs) (Feero and Pande 2008), where links in different dimensions (e.g., horizontal wires vs. vertical Through-Silicon Vias or TSVs) can have distinct failure probabilities, the PEF model partitions the total fault set F by logical dimension: $F = F_0 \cup F_1 \cup \cdots \cup F_{n-1}$. It then applies a separate upper bound, $|F_i|$, to the number of faults in each dimension. This more nuanced approach led to a substantial breakthrough, with Zhuang et al. (2023b) using it to elevate the proven fault tolerance of Q_n^k from a linear function of its dimension n to an exponential one. Despite this advance, the PEF model's core limitation is that it partitions faults by *logical dimension*, not *physical locality*, failing to capture the underlying physical cause of spatially correlated failures.

The Gap and Other Advanced Models

The progression from the FT model to the PEF model highlights a persistent gap in fault tolerance research: the need to accurately model spatially correlated failures. Real-world failure data from large-scale systems consistently shows that failures are not independent but are clustered due to shared physical dependencies like power, cooling, or physical proximity (Hacker, Romero, and Carothers 2009; Vishwanath and Nagappan 2010; Pinheiro, Weber, and Barroso 2007; Xia et al. 2020).

In parallel to the PEF model, researchers have explored other advanced conditional and *structure-based* fault models, where failures are assumed to affect entire subgraphs rather than just independent components. For instance, recent studies have investigated the Hamiltonian properties of k-ary n-cubes when faults manifest as vertex-disjoint stars ($K_{1,s}$) (Sabir et al. 2023), and have proposed generalized metrics for structure connectivity in related cubic networks (Sabir and Lin 2025). Other advanced conditional measures, such as g-extra connectivity (Zhang and Meng 2021), component connectivity (Liu, Meng, and Sabir 2023), ℓ-connectivity (Gu et al. 2022), and mixed connectivity (Gu, Shi, and Fan 2017), also aim to avoid the pathological scenarios of the FT model by imposing constraints on the size or number of components in the surviving graph.

While these approaches represent a significant step forward, they are often constrained to specific, uniform subgraph shapes or focus on properties of the remaining graph rather than the topology of the fault region itself. This leaves a critical gap for a model, like RBF, that can directly represent the *spatial correlation* and *physical clustering* of faults as its primary modeling primitive.

Comparative Analysis Summary

To crystallize the distinctions between these models, Table 2 provides a direct comparison of their core assumptions, limitations, and key features. This table underscores the progressive increase in model realism and highlights the unique contributions of the RBF model.

Table 2: Comparative analysis of fault tolerance models.

Feature	Traditional Fault (FT) Model (Yang, Tan, and Hsu 2007)	Partitioned Edge Fault (PEF) Model (Zhuang et al. 2023b)	Region-Based Fault (RBF) Model (This Work)
Core Assumption	Faults are independent and randomly distributed.	Faults are partitioned by logical dimensions.	Faults are spatially correlated and form connected clusters.
Fault Distribution	Uniform; worst-case analysis (all faults can be adjacent).	Constrained per dimension ($\|F_i\|$).	Constrained by cluster size, shape, and separation distance (d_{sep}).
Model Realism	Low; unrealistic for power outages or thermal events.	Medium; captures dimensional variance, not physical proximity.	High; directly models real-world clustered failures (e.g., rack failures).
Key Limitation	Overly pessimistic fault tolerance estimates ($O(n)$).	Ignores physical locality; a single physical event can span dimensions.	Relies on a predefined set of cluster shapes ($\mathcal{S}$) in current version.
Tolerance Metric	Total number of faults, $\|F\|$.	Per-dimension fault counts.	Geometric dispersion of clusters, captured by $k_{\max}$ and d_{sep}.

B Formal Graph-Theoretic Definitions

This appendix provides the comprehensive graph-theoretic definitions referenced in the main text. This paper adheres to standard graph theory terminologies and notations, as found in references such as (Bondy and Murty 1976; Tutte 2001; Zhang and Chartrand 2006; Gould 2012; Gross, Yellen, and Anderson 2018), unless otherwise specified.

A graph is denoted as a pair $G = (V(G), E(G))$, where $V(G)$ is the set of vertices (nodes) and $E(G)$ is the set of edges connecting pairs of vertices. The total number of vertices and edges are given by their set cardinalities, $|V(G)|$ and $|E(G)|$.

A graph S is a subgraph of G if $V(S) \subseteq V(G)$ and $E(S) \subseteq E(G)$. For any subset of vertices $M \subseteq V(G)$, the induced subgraph $G[M]$ consists of the vertex set M and all edges in $E(G)$ with both endpoints in M.

To represent a network with link failures, we define a faulty graph $G - F$ by removing a set of faulty edges $F \subseteq E(G)$ from the original graph, resulting in the edge set $E(G) \setminus F$ while keeping the vertex set $V(G)$ intact.

A path, denoted $P = (v_0, v_1, \ldots, v_p)$, is a finite sequence of distinct vertices where each adjacent pair (v_i, v_{i+1}) for $i \in \{0, 1, \ldots, p-1\}$ corresponds to an edge. The length of such a path is p, the number of edges it contains.

A Hamiltonian path is a path that includes every vertex in the graph exactly once. A graph that contains a Hamiltonian path between every pair of distinct vertices is known as a Hamiltonian-connected graph.

C Practical Guidelines for RBF Parameterization

The practical utility of the RBF model hinges on the judicious selection of its parameters $(k_{\max}, s_{\max}, d_{\text{sep}}, \mathcal{S})$, a process informed by both theoretical requirements and empirical system characteristics. The theoretical underpinnings of our Hamiltonicity proof necessitate adherence to the feasibility constraint established in Theorem 1 and non-degeneracy conditions. Beyond these formal requirements, the parameters serve as a bridge to the physical reality of the target system. An analytical approach to parameterization involves mapping system architecture to model variables. For instance, $k_{\max}$ can correspond to the number of independent failure domains (e.g., power distribution units or cooling zones). To determine $s_{\max}$, one must analyze the impact of a single-point failure on network connectivity. For example, by analyzing a server rack, one would determine the maximum number of interconnected links—both internal to the rack and connecting to the wider network—that could be severed simultaneously. This value, representing the worst-case impact of a single failure event, provides a direct, empirical basis for setting $s_{\max}$. Similarly, d_{sep} should be calibrated to the physical or logical separation between these domains. The set of allowed shapes, $\mathcal{S}$, provides a topological vocabulary to describe observed fault patterns. In scenarios lacking historical failure data, an iterative calibration process, such as cross-validation, can be employed, beginning with conservative estimates that are progressively refined through system monitoring and analysis.