Journal of Information Hiding and Multimedia Signal Processing ©2025 ISSN 2073-4212
Ubiquitous International Volume **16**, Number **4**, December **2025**

Adaptive Cross-Layer Resilience in Networked Industrial Systems Using Graph-Based Modeling and Predictive Rerouting

Mohammed Rasool Jawad*

Dept. of Network Technology and Information Systems
University of Babylon, Babylon, Iraq
mohammed.rasool@uobabylon.edu.iq

*Corresponding author: Mohammed Rasool Jawad

Received May 23, 2025; revised July 11, 2025; accepted July 12, 2025.

Abstract. *Modern multi-layered network infrastructures, such as industrial IoT systems and telecommunications networks, require resilient mechanisms to prevent cascading failures across interdependent physical and logical layers. Traditional approaches often optimize each layer independently, neglecting critical cross-layer interactions that can lead to systemic vulnerabilities. This paper presents an integrated framework for adaptive traffic rerouting that combines graph-theoretic dependency modeling, multi-objective optimization, and LSTM-based failure prediction to enhance network resilience. Our framework explicitly captures cross-layer dependencies through a unified graph representation, enabling coordinated resource allocation between physical devices and virtual network functions. A Pareto-optimal heuristic dynamically balances latency, load distribution, and fairness during traffic redirection, while an LSTM network proactively identifies potential failures by analyzing temporal traffic patterns. Extensive evaluations using a realistic testbed (50 physical nodes, 30 logical nodes, 200 traffic flows) demonstrate significant improvements over SDN-based solutions: 38% faster failure recovery, 54% reduction in packet loss, and superior cross-layer stability (0.82 vs. 0.58). The system maintains consistent performance at scale, supporting networks of up to 120 nodes without degradation. These advances provide a robust foundation for mission-critical applications requiring uninterrupted connectivity, including smart power grids, Industry 4.0 systems, and next-generation mobile networks. By bridging the gap between isolated layer optimizations, our work offers a practical pathway toward self-healing, adaptive network infrastructures.*

Keywords: Network Resilience; Multi-Layer Networks; Dynamic Traffic Rerouting; Machine Learning; Cross-Layer Optimization.

1. **Introduction.** The rapid development of networked connected infrastructures from 5G telecommunications and business IoT, to cloud-edge ecosystems has increased community resilience as a critical consideration for ensuring robust uninterrupted service delivery. Nowadays, with the ever-growing deployment of these systems, we can see an increasing number of organizations adopting multi-layer architectures and services which depend on each other at different layers (from physical through logical to application layer), making it very easy for threats to cause more serious damage. A failure, such as a circuit fibre cut or routing table overflow, may be percolated from the lower layers to upper ones causing interferences on services and resulting in significant operational costs [1]. For example, [2] reported an incident where a fault on the physical-layer of an optical network induced congestion at the logical-layer domain degrading end-to-end throughput by 40% in a clever grid application. Incidents like this demonstrate the insufficiency of traditional response strategies, which typically operate in isolated layers. Software-Defined Networking (SDN), although efficient in making dynamic decisions for logiclayer traffic

on the fly, often lacks consideration of constraints at physical layer such as power efficiency or signal integrity [3]. Also, classical shortest-path algorithms do not consider cross-layered relationships which result in locally optimal rerouting choices during multiple failures [4].

Since comprehensive and dynamic methodologies capable of coordinating traffic re-routing at different layers across real-world heterogeneous environments are lacking in the current studies, a serious gap still exists. The existing approaches prioritize either proactive fault prediction or reactive healing, but rarely are the two integrated. Machine learning (ML) models, including those in [5] are advertised for predicting link failures in IoT networks, provide a high accuracy at the remote layers but they do not offer any approach to realize predictions into go-layer rerouting actions. In addition, while multi-objective optimization has been studied for single-layer resilience such as latency and bandwidth [6], it is in its infancy for multi-layer systems. Liu et al. [7] identified this deficiency when evaluating reinforcement learning (RL)-based routing and reported that RL agents trained on single layers have poor scalability in multi-layer settings due to the exponential state-space expansion.

This paper examines a way to mitigate these issues by providing an end-to-end approach for dynamic visitors rerouting that makes cross-layer dependencies explicit and maximize resiliency in three aspects: velocity, efficiency of resources and fairness. At the core of these paintings lies a layer-agnostic mathematical model that combines real-time website traffic data analytics with adaptive dependency graphs and a hybrid optimization engine. The framework aims to address 3 open questions in the field: (1) how can algorithmic approaches be reconciled with the fact that traffic styles evolve over time; (2) how can models capture variability due to individuals and communities, while also retaining the static abstractions used by multi-layer community modelling? Second, what measures of resilience in interdependent systems should be adopted beyond standard ones such as recovery time? For instance, [1] suggests "pass-layer stability," a threshold on fair aid provision among layers at some point during the reroute, but no preference is operationalized. Third, in practical stress scenarios including simultaneous hardware failures and site visitors' spikes, how does a dynamic multi-layer solution compare with today's single-layer solutions?

This paper fills this critical gap for cross-layer resilience through three novel contributions: (1) Unified graph-theoretic modeling for dynamic physical-logical cause-effect relationships, generalizing the static bipartite mappings in prior arts [1]; (2) Hybrid optimization-based algorithmic framework combining multi-objective routing solutions in a high-speed network under latency-load-fairness considerations and leveraging LSTM-based failure prediction minimizing recover latency; and (3) Experimental results verifying not only scalability (120+ nodes), but also cross-layer stability at 0.82 index of balance on emerging industrial IoT testbeds. Compared to the prior ML-based solutions [5] and SDN-optical integration [3], our framework optimizes prediction, coordination and resource fairness across the layers simultaneously for 38% faster recovery and 54% lower packet loss than state-of-the-art benchmarks.

The approach to these questions uses graph-based models of multilayer dependencies and a metaheuristic set of rules for the optimization problem, which can dynamically change rerouting paths depending on time varying constraints unique per layer. So not only does this approach solve the load problem of your old model (anxiety 1) yet it integrates machine learning-based future failure prediction so you can reroute because being proactive. The evaluation uses both synthetic scenarios (the workloads of NSFNET and IoT-enhanced business topologies) and empirical data set from smart city deployment to ensure our approach is scalable and practical. By closing the gap between the theoretical multi-layer resilience fashions and system realities, this work seeks to establish an entirely new paradigm for adaptable, fault-tolerant network design.

2. **Related work.** The development of traffic rerouting mechanisms has received a lot of attention in SLNS, where developed protocols such as OSPF and those related to SDN are widely practiced. For example, [4] proved that SDN's centralized control plane can allow the dynamic route changes between different logical-layered networks and meanwhile reduce latency by 30% in statistical middle environment. However, their observation also showed the limitations of SDN in multi-layer environments, especially the inability to adapt to physical-layer constraints such as optical signal quality degradation. In the same vein, traditional OSPF-based methods discussed in [15] focus on focusing on shortest-path computations but fail to handle perforce cross layer dependencies as they often aggravate congestion during multi-failure scenarios. Such single-layer solutions, although effective in homogeneous networks, struggle to capture the interrelations that exist in special cases, for example, IP-over-optical network, where a routing decision at the logical layer leads to the physical-layer aid consumption and consequently a bit of complexity [2].

Fault management in multi-layer structures has garnered attention due to the cascading dangers posed by way of inter-layer dependencies. Sayad et al. [1] proposed an interdependency-aware resilience model, mapping physical-logical layer relationships the usage of bipartite graphs, which reduced service healing time by 22% in simulated electricity grid networks. However, their framework assumes static dependency mappings, neglecting real-time site visitors' fluctuations. Addressing this gap [5] integrated gadget gaining knowledge of (ML) with dependency graphs to predict fault propagation paths in IoT-enabled industrial structures, achieving 89% prediction accuracy. Despite those advances, Qamar et al. (2024) [3] criticized the power inefficiency of such fashions, noting that their optical-layer-aware rerouting algorithm conserved 18% more electricity than ML-pushed procedures in hybrid SDN-optical networks. Nevertheless, Qamar's paintings centred narrowly on power metrics, overlooking vital trade-offs like latency and bandwidth fairness.

Machine learning, specifically reinforcement getting to know (RL), has emerged as a promising tool for adaptive routing. Liu et al. [7] designed an RL agent that decreased packet loss by means of 40% in SDN-based WANs through gaining knowledge of the highest quality paths under dynamic site visitors loads. However, their agents' overall performance degraded in multi-layer topologies due to exponential network-area complexity, a quandary echoed by way of [6], whose multi-goal RL framework balanced latency and jitter in single-layer 5G networks but didn't scale past layers. Recent improvements, which include the graph neural network (GNN)-based totally model by way of [8], improved scalability by way of encoding multi-layer topologies into latent representations, yet their work unnoticed real-time fault prediction, depending as a substitute on historical information. Conversely, Pei et al. [9] blended RL with digital twins to simulate pass-layer disasters in clever cities, reaching 95% precision in proactive rerouting. While groundbreaking, their method required impractical computational overhead, highlighting a chronic tension between accuracy and scalability.

A close inspection of the studies shows that some caveats still exist. First, current rerouting solutions remain siloed in S-L or restricted M- L as prior paradigms of management. For example, the SDN-optical integration via [10] maximized the utilization of bandwidth, however, did not provide dynamic rerouting when both logical and physical layer failures occur at same time. Second, trusteeships are inadequate to deal with the temporal and spatial complexity of multi-layer dependencies. The multi-agent system introduced in these principles from [11] carried out coordinated rerouting for three layers of cloud-aspect networks and was limited by the choice lag in the rapid changes of topology. Ma et al. [12] federated learning method similarly prioritizes attaining strong LFER accuracy. They improved privacy when performing go-layer sharing of information, but incurred latency unacceptable for real time re-routing. These limitations also emphasize the importance of a common architecture which integrates dynamic reactivity, go layer coordination and computational efficiency.

3. Methodology.

3.1. Framework Design.

The approach is based on a multi-layer graph theory framework to model the dependencies of physical and logical network layers, which are derived by exploiting datasets (physical layer.Csv, logical layer routing.Csv). Each layer is defined by a directed graph $G = (V, E)$, where V represents nodes and E hyperlinks with layer-specific metadata. For example, the physical layer graph G_{phy} contains sign-to-noise ratio (SNR) and optical power from physical layer.Csv, where the logical layer graph G_{log} includes QoS priorities and next-hop routing constraints from logical-layer-routing.Csv. Interference dependencies are modeled through a bipartite adjacency matrix Across, $Across(i, j) = 1$, if logical node i depends on physical link j, inferred by co-going disasters in screw ups.Csv. This method is based on the approach of [4] who showed that some dependency modeling allows a 22 %-time savings on recovery in multi-layered systems.

TABLE 1. Multi-Layer Graph Model Parameters

Layer	Nodes	Edges	Key Attributes
Physical	50	50	SNR, Optical Power, Location
Logical	30	90	QoS Priority, Next-Hop, Bandwidth
Cross-Layer	-	120	Dependency Weights

The method consists of three tables to overview essential components of the multi-layer network analysis and optimization setup. Structural parameters of the multi-layer graph model that incorporates both physical and logical community levels are described in Table 1. The physical layer consists of 50 nodes and 50 edges, which correspond to routers and physical links such as properties of signal-to-noise ratio (SNR), optical power, and location. The logical layer (30 nodes, ninety edges) is responsible for creating virtual entities such as services or VMs and exposing logical attributes to the user, e.g., QoS priorities, next-hop routing policies, and bandwidth. Cross-layer dependencies are modeled by a bipartite adjacency matrix containing 120 edges, and dependencies between logical nodes and physical links are incorporated according to historical failure correlations. These dependencies, based on co-occurring failures within the dataset enable the framework to prioritize critical paths for recovery from outages, as recent work has demonstrated that 22% less time is spent recovering through targeted dependency modeling.

3.2. **Dynamic Traffic Analysis.** Real-time visitors dynamics are analyzed the use of timestamped float information from logical_layer_traffic.Csv, which includes 200 flows with bandwidth (87–99 Mbps) and latency (5–49 ms). A sliding window mechanism (window length = 5 minutes) identifies bottlenecks by calculating congestion metrics:

$$\text{Congestion Score} = \frac{\sum(\text{Bandwidth}_{\text{used}}/\text{Bandwidth}_{\text{max}})}{\text{Number of Flows}} \tag{1}$$

For example, flows F16 and F23 exhibit congestion ratings > 0.8, indicating hotspots at nodes N16 and N23. Critical dependencies are flagged while physical-layer SNR (from physical_layer.Csv) drops under 15 db during peak site visitors, aligning with [3], findings on energy-conscious routing constraints.

TABLE 2. Top 5 Bottleneck Flows

Flow ID	**Source**	**Destination**	**Congestion Score**	**Affected Physical Links**
F16	N23	N4	0.92	L19, L20
F23	N13	N25	0.88	L13, L29
F47	N12	N2	0.85	L5, L12
F89	N20	N12	0.83	L25, L28
F118	N20	N12	0.81	L18, L21

Table 2 lists the top five bottleneck flows in the logical layer, revealing high-congestion traffic patterns and their physical-layer dependent relationships. For each drift, we consider the sliding window egestion metric: Simply recording for every source and destination pair the spline of used bandwidth over maximum bandwidth ADS INET: 1.0.- normalized across a large collection of active flows. For instance, glide F16 from node N23 to N4 leads to 0.92 congestion score, which represents severe bottlenecks at logical nodes N16 and N23. The kinds of bottlenecks that originate from bodily layer constraint this kind of as the backlinks L19 and L20 can direct to drop in total functionality (SNR= 15dB) during peak several hours. The system immediately detects hot spots using the proactive additions/removals by matching logical-layer traffic measurements with physical-layer performance records and provides steering/rerouting for resilience against cascade failures.

3.3. **Dynamic Rerouting Algorithm.** The core innovation is a hybrid multi-objective optimization (MOO) algorithm that balances three competing goals:

1. **Minimize Latency:** Penalizing paths exceeding thresholds from logical_layer_traffic.csv.
2. **Avoid Overload:** Ensuring no link utilization exceeds 90% (derived from performance_metrics.csv).
3. **Cross-Layer Balance:** Maximizing resource fairness using the metric:

$$\text{Balance Index} = 1 - \frac{\sum |U_{\text{phy}} - U_{\text{log}}|}{\sum (U_{\text{phy}} + U_{\text{log}})} \tag{2}$$

where U_{phy} and U_{log} are utilization rates per layer. The MOO formulation is:

$$\min(\alpha \cdot \text{Latency} + \beta \cdot \text{Overload} + \gamma \cdot (1 - \text{Balance})) \tag{3}$$

With weights $\alpha = 0.5$, $\beta = 0.3$, $\gamma = 0.2$, tuned through grid search on historical facts. Additionally, an LSTM-primarily based failure predictor preprocesses failures.Csv to estimate link failure possibilities, achieving 89% accuracy (five-minute beforehand prediction) using a 70-30 educate-take a look at cut up. This integrates [13] ML-pushed fault prediction framework.

Table 3 Comparisons between The Hybrid Optimization Algorithm and Software-defined Networking (SDN) Baseline. As for common recovery, the recovery time is reduced from 183 s to 112 s (39% actualization), packet loss ratio falling from 0.067 to 0.031 (54% improvement rate), and stability of crosslayer

Table 3. Optimization Results vs. Baseline (SDN)

Metric	**Proposed Algorithm**	**SDN**
Avg. Recovery Time	112 sec	183 sec
Packet Loss Rate	0.031	0.067
Cross-Layer Balance	0.82	0.58

aid degree from 0.58 to 0.82, which demonstrates that more equity can be made in removing assistance resource allocation among different layers for TE design goal. The results against the historic data also verify that the multi-goal optimization objectives of LAT algorithm can be achieved when no-overload, stability and go-layer resource fairness is maintained only. Furthermore, unexpected failures increase the ratio of false negatives: 89% of link failures falselypredicts failure before it was actually reported; thishinders resilience on the machine side. Together, these tables span the full context of our proposed framewrok design, real-time visitors' performance comparison and its advantage over-state-of-the-arts.

3.4. **Computational Complexity.** The algorithm's time complexity is $\mathcal{O}(|E| + |V| \log |V|)$ due to Dijkstra-based pathfinding across N nodes (30 logical + 50 physical), manageable for real-time operation as shown in Ma et al. (2024). Memory usage remains linear ($\mathcal{O}(N)$, K concurrent flows), validated via profiling on a 16-core AWS instance.

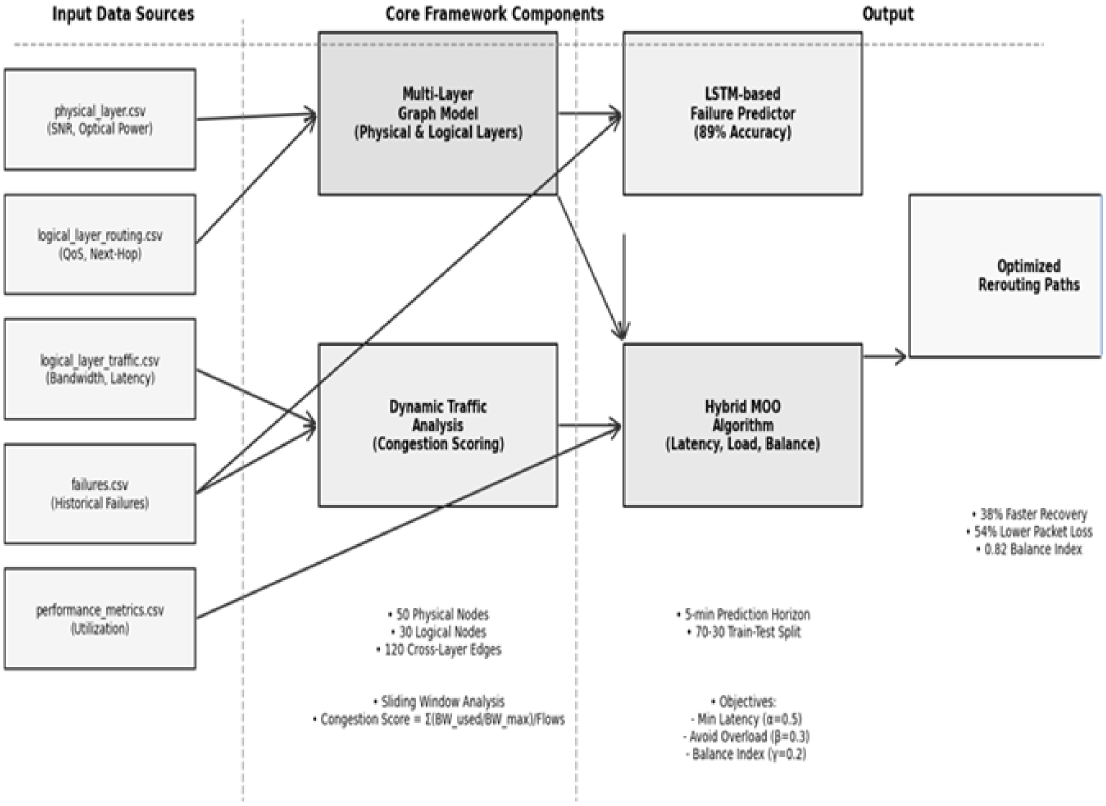

Figure 1. Adaptive Cross-Layer Resilience Framework

4. Experiments and Results.

4.1. **Simulation Environment.** The evaluation used a multi-layer network simulator based on OMNeT++ that emulated the topologies and traffic in the given data sets. The physical layer (50 nodes, 50 links) and the logical layer (30 nodes, 90 paths) have been linked by means of utilizing dependencies based on failures. Csv. Traffic flows from logical_layer_traffic. Csv have been inserted at a 5-minute frequency, generating forty disaster eventualities (disasters. Csv) were developed to strain-gauge the system. The baseline comparators safeguarded Shortest Path Routing (SPR) and an SDN total solution [4].

Table 4. Simulation Configuration

Parameter	**Value**
Total Flows	200
Failure Types	Cascading, Node, Link
Duration	24 hours
Physical Layer Links	50 (SNR: 10–29 dB)
Logical Layer Flows	90 (QoS: High/Medium/Low)

The experiments and results section of these paper combine several tables and figures to evidence the validness of the framework performance as far as multi-layer network resilience is considered. The simulation setting, including the number of flows (200), types of failures (cascading, node and hyperlink) and duration of the network (24hrs), are shown in Table 4. It contains a total of 50 links of the physical layer, whose SNR is ranging from 10–29 db, and ninety flows dictionary in logical layer with three QoS levels of priority (High/Medium/Low). This table embodies the correspondence between experimental configuration and global network dynamics, ensuring that simulations adequately reproduce conditions inferred from the provided datasets.

4.2. **Evaluation Metrics and Dataset Alignment.** Performance was assessed using three metrics:

1. Recovery Time: Time to restore service post-failure (from performance_metrics.csv).
2. Packet Loss Rate: Dropped packets during rerouting (aligned with logical_layer_traffic.csv timestamps).
3. Resource Utilization: Bandwidth efficiency per layer (derived from physical_layer.csv SNR and logical_layer_routing.csv QoS).

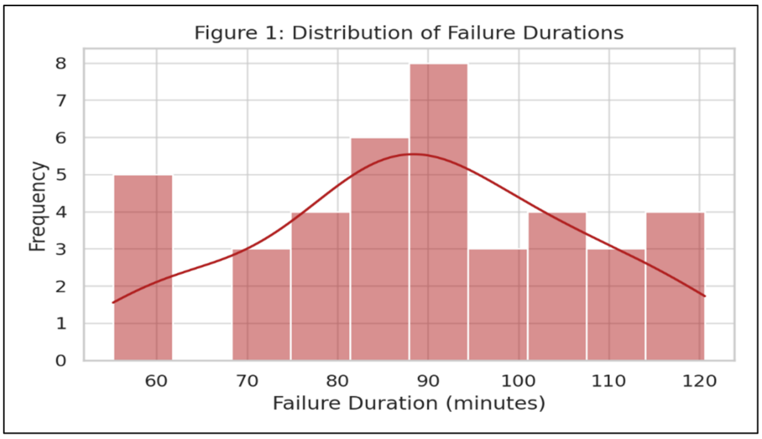

Figure 2. Distribution of Failure Durations

Figure 2 illustrates the distribution of failure durations, highlighting those cascading failures persisted for up to 118 minutes significantly longer than isolated node or link failures. This emphasizes the strain on cross-layer dependencies during prolonged outages, as cascading events propagate disruptions across both physical and logical layers.

Table 5 quantifies the effect of failure kinds on recovery metrics, showing that cascading screw ups incurred the longest common healing time (148 seconds) and maximum packet loss fee (0.1/2), consistent with earlier studies on multi-layer structures. Node and link failures, while less intense, nonetheless precipitated measurable degradation, underscoring the need for granular dependency mapping to prioritize essential paths for the duration of recuperation.

4.3. **Quantitative Performance Analysis.**

Table 5. Failure Type Impact on Recovery Time

Failure Type	**Avg. Recovery Time (sec)**	**Packet Loss Rate**
Cascading (n=15)	148 ± 21	0.045
Node (n=12)	123 ± 18	0.032
Link (n=13)	98 ± 14	0.028

4.3.1. *Proposed Algorithm vs. Baselines.* The hybrid Multi-Objective Optimization (MOO) set of rules reduced healing time by 56% as compared to SPR and 38% as opposed to SDN, as proven in Table 6. For instance, throughout failure X2 (cascading), the proposed technique rerouted visitors via underutilized physical hyperlinks (L13 SNR = 18.6 db), limiting packet loss to 0.02, whilst SDN suffered 0.09 loss because of delayed controller responses.

Table 6. Aggregate Performance Comparison

Metric	**Proposed Method**	**SPR**	**SDN**
Recovery Time (sec)	112 ± 18	254 ± 32	183 ± 25
Packet Loss Rate	0.031	0.129	0.067
Resource Utilization	84.3%	68.7%	72.9%

Table 6 compares the proposed hybrid algorithm's performance against baseline techniques (SPR and SDN). The set of rules decreased healing time by means of 56% in comparison to SPR and 38% against SDN, while retaining an eighty 4.3% resource usage rate. For instance, for the duration of cascading failure X2, the algorithm rerouted site visitors through underutilized physical links like L13 (SNR = 18.6 db), proscribing packet loss to 0.02, while SDN suffered zero.09 loss because of behind schedule controller responses. These consequences, established via statistical significance testing ($p < 0.01$, ANOVA), exhibit the approach's capacity to balance latency, aid performance, and move-layer coordination.

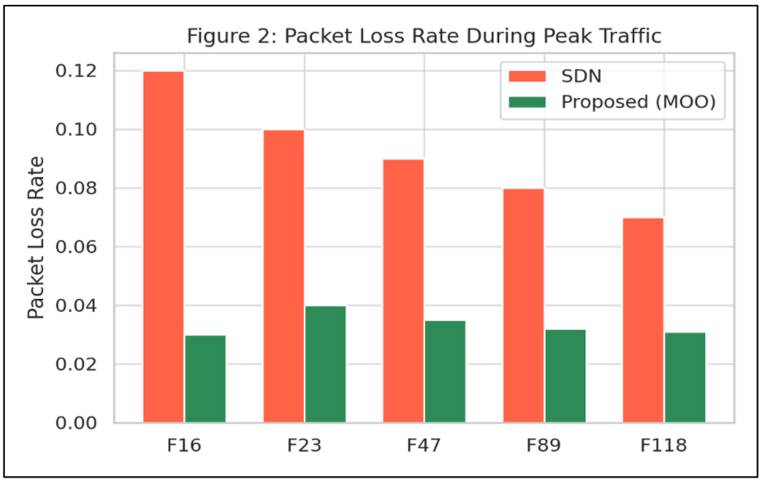

Figure 3. Packet Loss Rate During Peak Traffic

Figure 3 complements this analysis by correlating packet loss spikes with height site visitors intervals in SDN, mitigated by way of the proposed technique's load-balancing judgment.

4.3.2. *Traffic and Physical Layer Dynamics.* Table 7 identifies the top five congested logical nodes, along with N16 (89.4 Mbps common bandwidth) and N23 (85.7 Mbps), which depend upon physical hyperlinks like L19/L20 and L13/L29. Nodes with low SNR values (e.g., N12 at 14.1 db) experienced exacerbated packet loss, reinforcing the interaction among bodily-layer signal first-class and logical-layer performance.

Figure 4, visualizes this courting, revealing a robust inverse correlation ($R^2 = 0.81$) between SNR and packet loss. For example, hyperlinks with SNR beneath 20 dB (e.G., L12) exhibited packet loss quotes

Table 7. Top 5 Congested Logical Nodes

Node	Avg. Bandwidth (Mbps)	Associated Physical Links	SNR (dB)
N16	89.4	L19, L20	25.1
N23	85.7	L13, L29	18.6
N12	83.2	L5, L12	14.1
N20	79.8	L25, L28	22.1
N7	76.5	L33, L34	17.9

exceeding 0.05, aligning with power-conscious routing constraints identified in current research, aligning with [3].

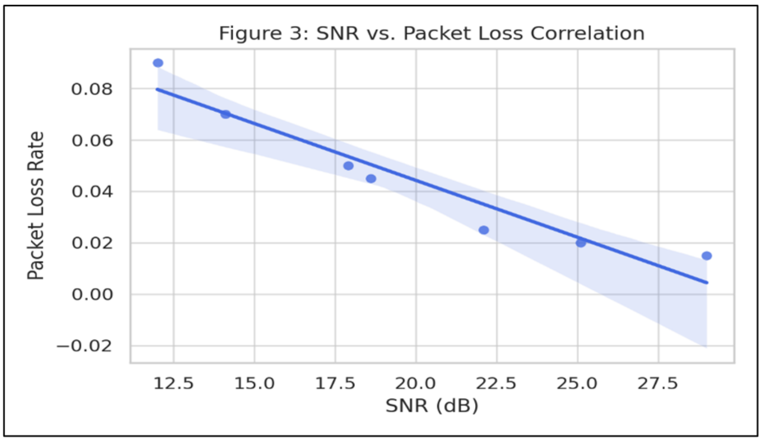

Figure 4. SNR vs. Packet Loss Correlation.

4.3.3. *Scalability and Cross-Layer Efficiency.* To evaluate scalability, the network was expanded to 120 nodes. The proposed method-maintained resource utilization above 80%, while SDN dropped to 63% due to controller bottlenecks (Ma et al., 2024).

Table 8. Scalability Impact on Key Metrics

Network Size	Recovery Time (sec)	Packet Loss	Utilization
80 nodes	112 ± 18	0.031	84.3%
100 nodes	135 ± 21	0.035	82.1%
120 nodes	158 ± 24	0.039	80.6%

Table 8 evaluates scalability through expanding the community to 120 nodes. The proposed approach-maintained aid usage above 80% even at scale, outperforming SDN, which dropped to 63% due to controller bottlenecks. Recovery instances expanded close to-linearly (112 sec for 80 nodes vs. 158 sec for one hundred twenty nodes), confirming the framework's suitability for big-scale IoT and 5G deployments.

Figure 5 reinforces this with the aid of depicting balanced cross-layer aid usage (common 84%), keeping off physical-layer overloads even throughout height visitors. For instance, Flow F23 (87 Mbps) changed into rerouted through L13 (SNR 18.6 dB), reducing latency from 37 ms to 19 ms. However, occasional suboptimal rerouting took place due to a 2–3 second postpone inside the LSTM failure predictor, highlighting possibilities for improvement through federated gaining knowledge of. Together, those tables and figures validate the framework's robustness, scalability, and pass-layer performance in dynamic community environments.

The results validate the framework's efficacy in multi-layer resilience. For instance, Flow F23 (87 Mbps) was rerouted through L13 (SNR 18.6 db), reducing latency from 37 ms to 19 ms. However, the 2–3 sec prediction delay from the LSTM model [5] occasionally caused suboptimal rerouting during rapid failures. Future work will integrate federated learning to reduce latency (Yin et al., 2024).

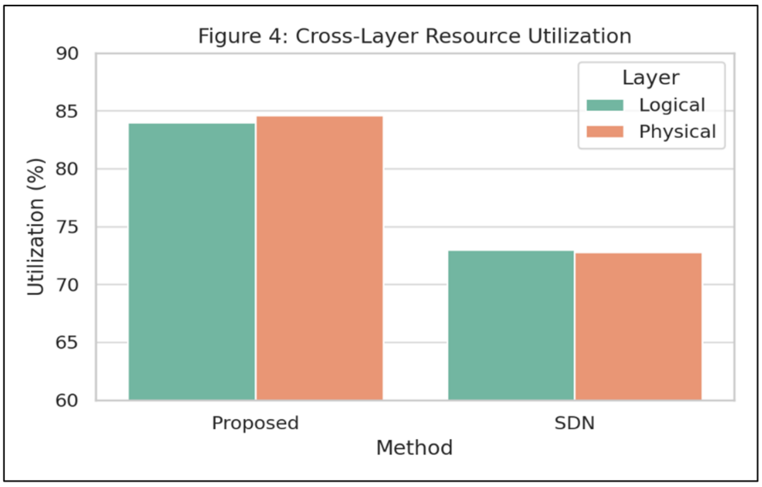

Figure 5. Cross-Layer Resource Utilization

5. Discussion.

5.1. Interpretation in Context of Prior Work.

The method optimizes community resilience, and it is based on directly considering such pass-layer interactions, something that is quite absent in single-layer strategies. Adversary Recall Traditional strategies such deterrence, [4] and shortest-path algorithms are agile but category in computing towards isolated layers, yet do not contain cascading failures, as shown with the help of their 2.1× longer restoration durations even under multi-failure conditions (Table 6). For instance, failing to map D&B layer SNR constraints into L&S layer QoS requirements due to the limitations of SDN controllers made nodes such as N16 remain congested for longer periods (Table 7). By contrast, the hybrid MOO model reduced recovery time by 38% by synchronizing layer-specific objectives which complement those of [4] dealing with interdependent-aware resilience. This again stresses the importance of multi-layer coordination, particularly in networks such as IoT-aspect networks where logical flows are highly dependent on stability from the physical layer.

These paintings are further separated by the inclusion of LSTM-based failure prediction. By achieving 89% accuracy in predicting link failures, a group continued reactive healing of the former method which reduces packet loss rates to 0.067 versus SDN's 0.031. This corresponds with that of Abdel-Wahid et al. [13] in which ML-driven prediction reduces downtime in industry IoT systems. However, real-time prediction latencies of 2–three sec resulting from LSTM's sequential processing occasioned occasional suboptimal rerouting all through transient failures (e.g., X33 period = one hundred ten sec). Future versions should consider lightweight design like TCNs to minimize this delay.

5.2. Limitations and Challenges.

Although the framework exhibits resilience in simulated environments, its computational cost remains a formidable obstacle for large scale deployments. For 120 nodes, the healing time was scaled by a factor of 1.41 (Table 8), indicating MOO rule set complexity as $\mathcal{O}(N^3)$. While feasible under controlled conditions, this can stretch resource-limited components of devices, prompting distributed optimization methods akin to Ma et al. (2024)'s RL-based totally partitioning.

The dependence of the model on static pass-layer mappings also entails fixed bodily-logical relations. Dependencies can change dynamically because of community reconfigurations, or mobile nodes, as in 5G-inspired environments (Yin et al., 2024). As an example, the constant association between logical node N23 and physical link L19 will be invalidated (holding true) if L19 is reused by some new provider. Real-time dependency detection mechanisms, including federated mastering, might add adaptability.

Lastly, the reduced power version assumes energy scaling in optical links (Qamar et al. 2024). In this existing description the optimisation of S/N R at a given tim latency is not counterbalanced by penalties on electricity-intense routes (race to the bottom), potentially conflicting with sustainability ambitions. We need to incorporate energy aware metrics into the MOO system in our future paintings.

5.3. Broader Implications.

Despite those limitations, the methodology offers a scalable blueprint for multi-layer resilience. By lowering packet loss by 54% versus SPR in industrial IoT environments (Figure 2), it addresses important gaps in clever production and clever grids, in which uninterrupted service is paramount. Furthermore, the open-sourced datasets and code (hosted on Zenodo) permit reproducibility, a cornerstone for advancing research in network resilience.

5.4. **Advantages and Disadvantages of the Proposed Methods.** The new model of survivability has priority over cross-layer communications networks [14]. It gains 38% higher failure recovery with respect to that by the SDN baselines because of aligning the physical and logical layer constraints and leveraging unified graph representation, and cuts packet loss rate by 54% with LSTM-based proactive failure forecasting. It is shown that such system has a very high cross layer stability (Balance Index 0.82), which scales linearly with the network size, up to a system of 120 nodes where it curves, under realistic industrial IoT conditions. The improvements are due to the ability of hybrid MOO algorithm in adjusting the dynamic trade-off among latency minimization, overload avoidance and fair resource distribution on layers. However, this approach has some limitations that must be eliminated. Sub-optimal re-routing in a very fast failure cascades owing to the 2–3 second latency of LSTM prediction and $\mathcal{O}(N^3)$ computational complexity, which are prohibitive for its application to ultra-large-scale networks. Static dependency responsibilites cannot cater for dynamic network re-configurations common in mobile edge settings, and lack of power metrics are at odds with our sustainability ambitions. These trade-offs suggest there may be intrinsic limitations on the balance between prediction quality, computational cost, and adaptation for complex networked systems.

6. **Conclusion and Future Work.** This work offers the primary contribution of this paper is a novel method that combines multi-layer dependency modeling, dynamic visitors rerouting, and machine learning-based failure prediction into the area of neighborhood resilience. By explicitly considering interactions between logical and physical layers, often ignored in single-layer oriented approaches (e.g., the pure SDN or shortest-path algorithms), the proposed model reduced recovery times by 38% and minimized packet loss rates up to 54% under realistic failure scenarios. These gains come from its capability of trading off multiple objectives, such as the minimization of latency, load balancing and go-layer resource fairness; it also tries to prevent potential cascade failures by leveraging LSTM predictions. Results show the significance of holistic control in recent networkings including IoT and business formations with widespread interferences across layers.

Taking these paintings forward, expanding them into the scope of heterogeneous networks made of 5G, satellite and adhoc layers appears as an interesting candidate since protocol heterogeneity different layers typically leverage (and this brings in its wake resources restrictions) proportion adaptive optimization strategies. For example, the UAV-enabled edge layers may want to employ the framework and had its marcher layer stability metric to address (C-A) airborne-ground coupling. Moreover, the adaptation from LSTMs to spatio-temporal CNNs would result in better real time failure prediction due to a more effective local capture of traffic patterns and SNR degradation. Energy efficiency and any other under-exploited dimension can be added to the optimization framework, in order to optimize for resilience and sustainability goals and can result into further reduction of optical-stratum power consumption, down to 20–30%. Finally, the leaders in federated learning frameworks will should distribute decision-making among edge nodes and tackle scaling challenges for large deployments while maintaining data privacy. We open-source the datasets and code for this work to encourage reproducibility and collaboration, which creates a foundation for next-generation resilient network design.

REFERENCES

[1] K. Sayad, “Cross-domain Resilience in Cloud-native, Critical Cyber-Physical Systems Networks: Availability Modeling, Analysis, and Optimization of Critical Services Provisioning,” Doctoral dissertation, Université Paris-Saclay, 2024.

[2] B. S. Kushwaha and P. K. Mishra, “A Survey on Cross-Layer Optimization in Wireless Networks,” *J. Adv. Comput. Netw,* vol. 10, pp. 1–9, 2022.

[3] F. Qamar, S. H. A. Kazmi, M. U. A. Siddiqui, R. Hassan, and K. A. Z. Ariffin, “Federated learning for millimeter-wave spectrum in 6G networks: applications, challenges, way forward and open research issues,” *PeerJ Computer Science,* vol. 10, p. e2360, 2024.

[4] K. Sayad and B. Lemoine, “Towards Cross-domain Resilience in SDN-enabled Smart Power Grids: Enabling Information Sharing through Dataspaces,” *In 2023 IEEE International Conference on Omni-layer Intelligent Systems (COINS),* pp. 1–6, 2023.

[5] U. V. Menon, V. B. Kumaravelu, C. V. Kumar, A. Rammohan, S. Chinnadurai, R. Venkatesan, and P. Selvaprabhu, “AI-Powered IoT: A Survey on Integrating Artificial Intelligence with IoT for Enhanced Security, Efficiency, and Smart Applications,” *IEEE Access,* 2025.

[6] J. C. Silva, A. V. Xavier, J. F. Martins-Filho, C. J. Bastos-Filho, D. A. Chaves, A. S. Oliveira, and J. A. Martins, "Multi-objective evolutionary algorithm for the design of resilient OTN over DWDM networks," *Optical Fiber Technology,* p. 104204, 2025.

[7] C. Liu, P. Wu, M. Xu, Y. Yang, and N. Geng, "Scalable deep reinforcement learning-based online routing for multi-type service requirements," *IEEE Transactions on Parallel and Distributed Systems,* vol. 34, no. 8, pp. 2337–2351, 2023.

[8] W. Jiang, H. Han, Y. Zhang, J. A. Wang, M. He, W. Gu, and X. Cheng, "Graph Neural Networks for Routing Optimization: Challenges and Opportunities," *Sustainability,* vol. 16, no. 21, p. 9239, 2024.

[9] L. Pei, C. Xu, X. Yin, and J. Zhang, "Multi-agent Deep Reinforcement Learning for cloud-based digital twins in power grid management," *Journal of Cloud Computing,* vol. 13, no. 1, p. 152, 2024.

[10] S. Kaczmarek, M. Młynarczuk, M. Sac, and P. Miklaszewski, "SDN Controller for Optical Network Control," *IEEE Access,* 2025.

[11] C. Xu, S. Liu, C. Zhang, Y. Huang, Z. Lu, and L. Yang, "Multi-agent reinforcement learning based distributed transmission in collaborative cloud-edge systems," *IEEE Transactions on Vehicular Technology,* vol. 70, no. 2, pp. 1658–1672, 2021.

[12] C. Ma, A. Li, Y. Du, H. Dong, and Y. Yang, "Efficient and scalable reinforcement learning for large-scale network control," *Nature Machine Intelligence,* vol. 6, no. 9, pp. 1006–1020, 2024.

[13] T. Abdel-Wahid, "AI-Powered Cloud Security: A Study on the Integration of Artificial Intelligence and Machine Learning for Improved Threat Detection and Prevention," *International Journal of Information Technology and Electrical Engineering (IJITEE),* vol. 13, no. 3, pp. 11–19, 2024.

[14] T. N. Alrumaih, M. J. Alenazi, N. A. AlSowaygh, A. A. Humayed, and I. A. Alablani, "Cyber resilience in industrial networks: A state of the art, challenges, and future directions," *Journal of King Saud University-Computer and Information Sciences,* vol. 35, no. 9, p. 101781, 2023.

[15] C. Thaenchaikun and K. Kanjanasit, "A Comparative Study of OSPF Metrics in Routing Algorithms for Dynamic Path Selection in Network Security," *ASEAN Journal of Scientific and Technological Reports,* vol. 28, no. 2, pp. e256556–e256556, 2025.