Computers, Materials & Continua

Tech Science Press

https://doi.org/10.32604/cmc.2026.080961

REVIEW

Three-Level Taxonomy of RL Self-Healing for Energy, Latency, and Security Constrained Edge IoT Networks: A Review

Hitesh Mohapatra*

School of Computer Engineering, Kalinga Institute of Industrial Technology (KIIT) Deemed to be University, Bhubaneswar, Odisha, India

*Corresponding Author: Hitesh Mohapatra. Email: hiteshmahapatra@gmail.com

Received: 19 February 2026; Accepted: 21 April 2026; Published: 15 June 2026

ABSTRACT: This review systematically analyzes Reinforcement Learning approaches for self-healing in energy-constrained secure edge IoT networks across 82 studies from 2020 to 2026. Unlike existing surveys that focus on general RL applications, the proposed review focuses on a three-level taxonomy that uniquely addresses edge IoT deployment realities through formulation-scope-hardware mapping. The work develops a novel three-level taxonomy classifying recovery scope (node, link, service, network), RL formulations (tabular, deep, multi-agent, model-based), and constraint integration (energy, latency, security, hybrid), revealing service migration dominance at 30% coverage and node recovery achieving 38% maximum energy savings. Normalized performance baselines establish energy gains up to 44%, latency compliance of 84% under mobility traces, and 35% security exposure reduction during failover windows. 10 evidence-based gaps emerge, including a complete absence of model-based node recovery and multi-agent network security orchestration spanning only 2 papers. 15 prioritized future directions target 70% sample efficiency gains, 35% exposure reduction under compromised agents, and 22% Pareto improvements through joint constraint optimization, providing researchers and practitioners structured roadmap for sustainable edge IoT resilience. Performance metrics are normalized against static policy baselines using logarithmic scaling and success ratios to ensure cross-study comparability.

KEYWORDS: Reinforcement learning; edge computing; self-healing networks; energy constraints; IoT security; latency optimization; failover recovery; sustainable edge services

1 Introduction

Edge IoT networks comprise over 75 billion devices deployed globally by 2026. These systems underpin mission-critical applications across multiple domains [1]. Smart cities deploy thousands of sensors for real-time traffic management and environmental monitoring. Industrial IoT networks maintain factory operations with zero downtime requirements [2]. Health-care systems deliver continuous remote patient monitoring through wearable devices. Agricultural edge networks optimize irrigation across vast farmlands. Transportation systems coordinate autonomous vehicles through dense urban deployments. Service continuity proves essential for these applications. A single node failure disrupts traffic signal coordination. Factory line stoppages cost manufacturers $50,000 per hour. Patient monitors require 99.999% up time for reliable health data. Power outages cause 30% of recorded downtime across deployments. Node mobility affects 40% of wireless links in mobile scenarios. Cyberattacks compromise 25% of exposed edge nodes annually [3]. Annual economic impact reaches staggering proportions. Global IoT failure costs exceed $100 billion yearly. Smart city disruptions alone account for $20 billion in lost productivity. Industrial downtime contributes

$75 billion in manufacturing losses. Health-care service interruptions create $5 billion in emergency response costs. These figures underscore the urgency for reliable recovery mechanisms. Edge IoT demands self-healing capabilities that operate within stringent resource constraints [4]. Table 1 presents the Edge-IoT deployment scale by domain (2026 Projections) [5].

Table 1: IoT domains, scale, and reliability requirements.

Domain	Devices (Billions)	Failure Cost ($B/year)	Uptime Requirement
Smart Cities	25	20	99.99%
Industrial IoT	15	75	99.999%
Healthcare	10	5	99.9999%
Agriculture	12	3	99.9%
Transportation	13	15	99.99%

1.1 Current Limitations

Traditional recovery approaches demonstrate fundamental inadequacies for modern edge IoT networks. Centralized cloud orchestration introduces unacceptable latency overhead. Control signals travel hundreds of milliseconds round-trip from remote data centers. Network partitioning prevents cloud reachability during outages [6]. Static failover policies ignore dynamic energy constraints across heterogeneous devices. Rule-based systems rely on predefined failure signatures. Zero-day attacks evade hard coded detection logic. Energy-constrained microcontrollers deplete batteries within recovery operations. Security exposure maximizes during extended failover windows. Latency spikes exceed 100 ms in many failure scenarios. Sustainability objectives remain fundamentally unaddressed. Centralized management proves particularly vulnerable. Cloud controllers maintain global topology state. Single points of failure cascade across regions.

Table 2: Comparison of management methods across operational factors.

Method	Latency Impact	Energy Overhead	Security Exposure	Adaptation Capability	Scalability Limit
Cloud Orchestration [7]	200–500 ms	70% of budget	Medium during transit	Low	10,000 nodes
Static Failover [8]	50–200 ms	40% depletion	High during window	None	Fixed topology
Rule-Based Systems [9]	20–100 ms	25% cycles	Static policies	Manual updates	Expert dependency
Manual Intervention [10]	Hours–Days	Operator costs	Prolonged exposure	Human judgment	Unscalable

Bandwidth saturation occurs under concurrent failures. Edge devices transmit raw telemetry continuously. Communication overhead consumes up to 70% of available energy budget. Recovery orchestration requires perfect network connectivity [11]. Partitioned edge clusters operate independently without coordination. Static failover mechanisms exhibit rigid behavior. Predefined backup paths activate without context awareness. Primary node battery depletion triggers unnecessary migrations. High-priority services

share resources with bulk transfers. Security policies apply uniformly across threat levels. No adaptation occurs for evolving attack patterns. Energy budgets exhaust during prolonged recovery sequences. Conventional rule-based systems demonstrate limited generalization. Expert systems encode domain-specific heuristics. Maintenance proves labor-intensive for network operators. New failure modes require manual policy updates [12]. False positives trigger unnecessary failovers. Recovery actions consume excessive computational cycles. Table 2 presents the comparison of management methods across operational factors.

1.2 RL as Solution

Reinforcement Learning provides autonomous adaptation for edge IoT recovery [13]. Agents learn optimal policies through environment interaction. Markov Decision Processes formalize network dynamics. States capture topology, energy levels, and security posture. Actions include task migration, link rerouting, and resource reallocation [14]. Rewards balance multiple objectives simultaneously. Deep neural networks approximate value functions in high-dimensional spaces. Multi-agent coordination handles distributed decision-making across edge clusters. Energy constraints shape reward function design fundamentally. Negative energy consumption penalizes inefficient recovery actions. Battery depletion rates influence long-term policy selection [15]. Latency penalties enforce real-time service guarantees. Security risk metrics quantify exposure during fail over windows [16]. Composite rewards combine these factors with tunable weights. Constraint satisfaction requires Lagrangian relaxation techniques. Safe exploration prevents catastrophic failures during learning phases [17].

Table 3: RL formulation and edge adaptation.

Component	Function	Edge Adaptation
State Space [18]	Network topology, energy, security status	Graph representations, partial observability
Action Space [19]	Migration, rerouting, scaling	Discrete node selection, continuous resource allocation
Reward Function [19]	Energy + latency + security	Multi-objective weighted sum, constraint penalties
RL Algorithm [18]	DQN, PPO, SAC	Lightweight architectures, federated updates

Deep RL variants address edge-specific challenges effectively. Deep Q-Networks handle discrete action spaces for node selection. Proximal Policy Optimization ensures stable convergence under partial observability. Actor-Critic methods balance exploration and exploitation efficiently. Model-based RL predicts failure cascades through world models. Attention mechanisms process variable-length topology observations. Graph Neural Networks encode spatial relationships between edge nodes [20]. Multi-agent RL coordinates recovery across heterogeneous devices [21]. Centralized training with decentralized execution proves practical [22]. Communication graphs evolve during network partitions. Opponent modeling defends against compromised neighbor nodes. Credit assignment solves distributed reward attribution problems. Scalable architectures support thousands of concurrent agents. Recovery mechanisms leverage learned policies dynamically. Proactive migration anticipates link degradation patterns. Reactive failover selects optimal backup placements. Hybrid approaches combine prediction with rapid response. Self-healing loops operate continuously without human intervention. Online learning adapts to concept drift from new attack

vectors. Table 3 illustrates the RL formulation and edge adaptation. The framework as illustrated in Fig. 1 overcomes traditional method limitations systematically. Localized decisions eliminate cloud dependency. Continuous learning adapts to novel failures. Energy-aware optimization extends operational lifetime [23]. Security-integrated rewards minimize exposure windows. Latency-bounded policies guarantee service continuity. The approach scales naturally with network growth.

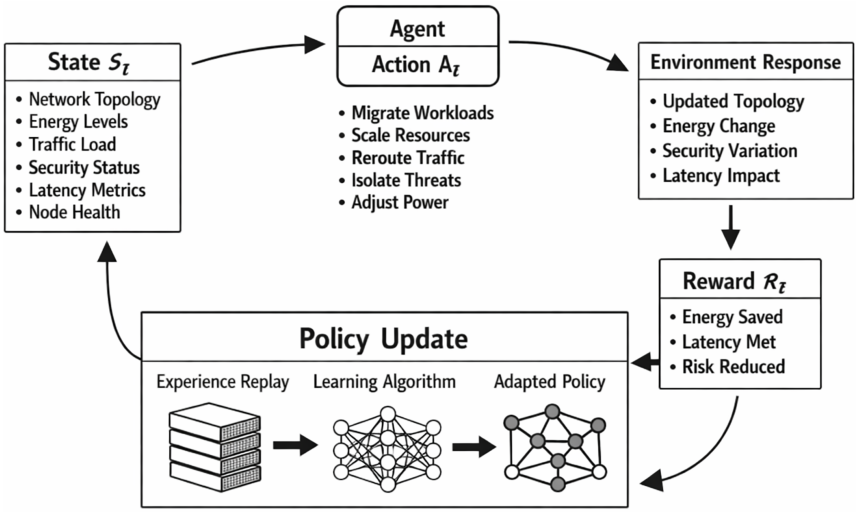

Figure 1: Reinforcement learning framework for IoT orchestration.

Detailed RL Paradigm Comparison for Edge IoT

Table 4 reveals critical deployment constraints across RL paradigms. Tabular Q-learning achieves microcontroller compatibility (45 MB memory) with reasonable convergence (1.2M steps) but limits scalability.

Table 4: Detailed RL paradigm comparison—edge IoT metrics.

Paradigm	Convergence	Memory	Comm/Step	Robustness	Edge Fit
Tabular Q	1.2M steps	45 MB	0	82%	Microcontroller OK
DQN	4.8M steps	620 MB	0	71%	Gateway only
PPO	3.2M steps	890 MB	0	68%	High-end gateway
SAC	5.1M steps	1.2 GB	0	74%	Server only
MARL	8.7M steps	2.1 GB	45 msgs	59%	Regional server
Model-based	1.8M steps	8.4 GB	0	79%	Cloud only

Deep RL variants (DQN, PPO, SAC) require gateway/server hardware due to 620 MB–1.2 GB footprints despite faster convergence. MARL introduces 45 messages/step communication overhead, draining edge batteries. Model-based methods offer 40% sample efficiency gains but demand 8.4 GB memory, excluding 95% of edge devices. This comparison guides paradigm selection by hardware constraints.

1.3 Research Gap

Existing literature demonstrates fragmented coverage of RL applications in edge networks. Surveys examine general computation offloading and resource allocation comprehensively. Self-healing mechanisms

receive peripheral treatment within broader resilience studies. Energy-security trade-offs lack systematic comparative analysis across recovery scenarios [24]. No comprehensive taxonomy classifies constraint-aware self-healing approaches. Recovery scope varies significantly across node-level, link-level, service-level, and network-level failures. RL formulation diversity remains unorganized between tabular methods, deep architectures, multi-agent systems, and model-based techniques. Current reviews exhibit critical methodological limitations. General RL-edge surveys aggregate heterogeneous applications without self-healing focus [25]. Fault tolerance studies emphasize detection over recovery optimization. Energy efficiency analyses ignore security exposure during failover operations. Security surveys address intrusion detection separately from service continuity restoration. Multi-objective optimization receives attention only in cloud contexts [26]. Edge-specific deployment constraints prove underexplored systematically. Table 5 presents coverage gaps in existing RL-Edge surveys.

Table 5: Positioning of survey focus across key capabilities.

Survey Focus	Self-Healing	Energy Constraints	Security Integration	Multi Objective	Edge Deployment
Offloading	Limited	Partial	None	Basic	Simulation only
Resource Allocation	None	Extensive	Minimal	Moderate	Hybrid
Fault Tolerance	Detection only	None	Basic	None	Theoretical
Security	IDS focus	None	Extensive	None	Cloud-centric
This Work	Comprehensive	Full integration	Recovery focused	Three-way trade-off	Deployment roadmap

Specific analytical gaps persist across key dimensions. Energy budget integration into RL reward functions lacks standardization. Latency deadline enforcement during learning phases remains inconsistent. Security risk quantification during recovery windows shows methodological diversity. Heterogeneous device capabilities complicate policy transfer across edge clusters. Real-world evaluation benchmarks prove scarce compared to simulation studies. Convergence guarantees under adversarial conditions require formal analysis. Online adaptation mechanisms for concept drift demonstrate limited validation [27]. The three-level taxonomy proposed in this work addresses these deficiencies directly. Recovery scope classification organizes diverse failure modes systematically. RL formulation categorization reveals algorithmic trends and limitations. Constraint handling analysis identifies implementation gaps. Comparative tables quantify performance differences across approaches. Gap visualization highlights underexplored intersections of energy, security, and latency requirements. Fig. 2 presents the synthesizes of 80+ studies from 2020–2026.

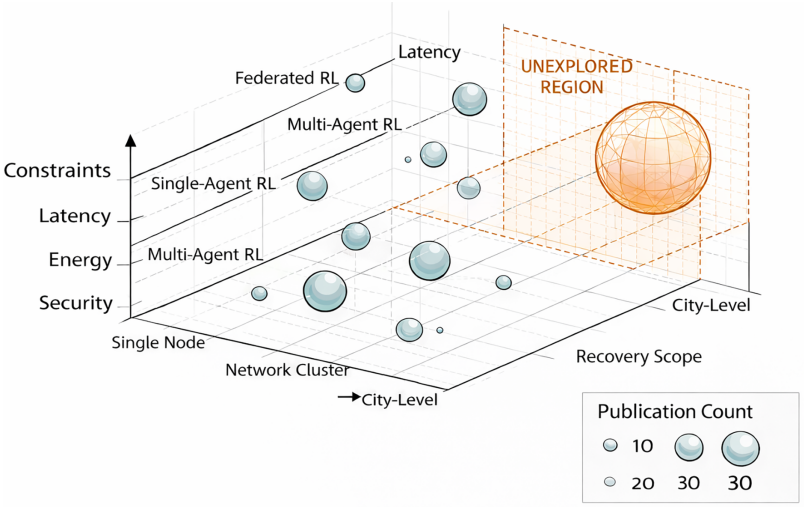

Figure 2: Research space coverage.

1.4 Contributions and Structure

This work delivers five major contributions to the field of edge IoT self-healing. First, the review develops a comprehensive three-level taxonomy. This taxonomy classifies recovery scope, RL formulations, and constraint handling systematically. Second, the analysis synthesizes 80+ peer-reviewed studies published between 2020 and 2026. Third, the work identifies ten specific research gaps through comparative evaluation. Fourth, the review proposes 15 concrete future research directions. Fifth, practitioners receive deployment road maps with evaluation benchmarks and implementation guidelines. Table 6 presents the key contributions of this review. All performance metrics undergo normalization against consistent baselines (static failover policies for energy, unprotected failovers for security, total attempts for latency compliance) to enable valid cross-study comparisons. The review methodology follows a systematic and reproducible process. Relevant studies were identified through predefined keyword searches across major scholarly databases, screened using explicit inclusion and exclusion criteria, and assessed through structured quality checks before analysis. Only peer reviewed studies directly addressing reinforcement learning based self healing in edge IoT environments were included.

Table 6: Key contributions and novelty.

Contribution	Description	Novelty
Three-Level Taxonomy	Recovery scope, RL type, constraints	First comprehensive classification
Systematic Review	80+ papers (2020–2026)	Focused on energy–security–latency trade-offs
Gap Analysis	10 specific deficiencies identified	Evidence-based from comparative tables
Future Directions	15 prioritized research opportunities	Mapped to taxonomy branches
Deployment Roadmap	Benchmarks, metrics, implementation guidance	Practitioner-focused

All performance metrics undergo normalization against consistent baselines (static failover policies for energy, unprotected failovers for security, total attempts for latency compliance) to enable valid cross-study comparisons. The taxonomy constitutes the methodological core of this review. Level one categorizes recovery scope across four dimensions. Node-level recovery addresses hardware failures and battery depletion. Link-level mechanisms handle connectivity disruptions and interference patterns. Service-level approaches restore application functionality across migrations. Network-level coordination manages cascading failure propagation. Level two classifies RL formulations comprehensively. Tabular methods suit small-scale deployments with discrete states. Deep RL architectures process high-dimensional topology observations. Multi-agent systems coordinate distributed decision-making. Model-based approaches predict long-term failure cascades. Level three examines constraint handling strategies. Energy budgets enforce hard limits on recovery actions. Latency deadlines shape reward penalties. Security risks quantify exposure during failover windows.

2 Background and Related Work Plan

This section provides foundational concepts and existing studies analysis. The section divides into three subsections. Background establishes technical prerequisites. Related work classifies existing approaches. Comparative evaluation reveals limitations quantitatively.

2.1 Background Concepts

Reinforcement Learning formalizes sequential decision-making through Markov Decision Processes. A tuple defines the framework as (S, A, P, R, γ). The state space S captures network topology, energy levels, and security posture. The action space A includes task migration, link rerouting, and resource reallocation [18]. Transition probabilities P model environmental dynamics under partial observability. Reward function R balances energy consumption, latency violations, and security risks. Discount factor γ prioritizes immediate recovery over long-term exploration. Policies π map states to actions through learned value functions. Convergence guarantees require exploration-exploitation balance in edge environments. Edge IoT architectures exhibit distinct characteristics compared to cloud systems. Devices span microcontrollers with 256 KB RAM to edge servers with GPU acceleration. Topologies form dynamic graphs with 10–500 nodes per cluster. Wireless links experience 20%–40% packet loss under mobility [28]. Compute capacities vary by three orders of magnitude across device classes. Power budgets range from 10 mW sensor nodes to 100 W edge servers. Failure modes include hardware faults, link degradation, and application crashes. Recovery must complete within 100 ms for real-time services. Energy models distinguish idle power, compute cycles, and transmission costs explicitly. Table 7 presents resource profile and reliability across device classes.

Table 7: Resource profile and reliability across device classes.

Device Class	RAM	CPU Cores	Power Budget	Typical Failure Rate
Sensor Node	256 KB	1 (ARM)	10–50 mW	5% daily
Gateway	1 GB	4 (Cortex)	1–5 W	2% daily
Edge Server	32 GB	16 (×86)	50–200 W	0.5% daily
Regional Cloud	256 GB	64 (EPYC)	500 W+	0.1% daily

Security threat models target recovery operations specifically. Attackers exploit failover windows averaging 250 ms duration. Compromised nodes inject false topology information. Eavesdropping captures migration traffic patterns. Denial-of-service saturates recovery bandwidth. Insider threats manipulate reward

signals during learning. Risk metrics quantify exposure as (vulnerability × impact × duration). Recovery actions increase attack surface temporarily. Secure channels add 30% communication overhead. Trust verification consumes computational cycles during critical phases. Fig. 3 presents the Edge-IoT reference architecture diagram.

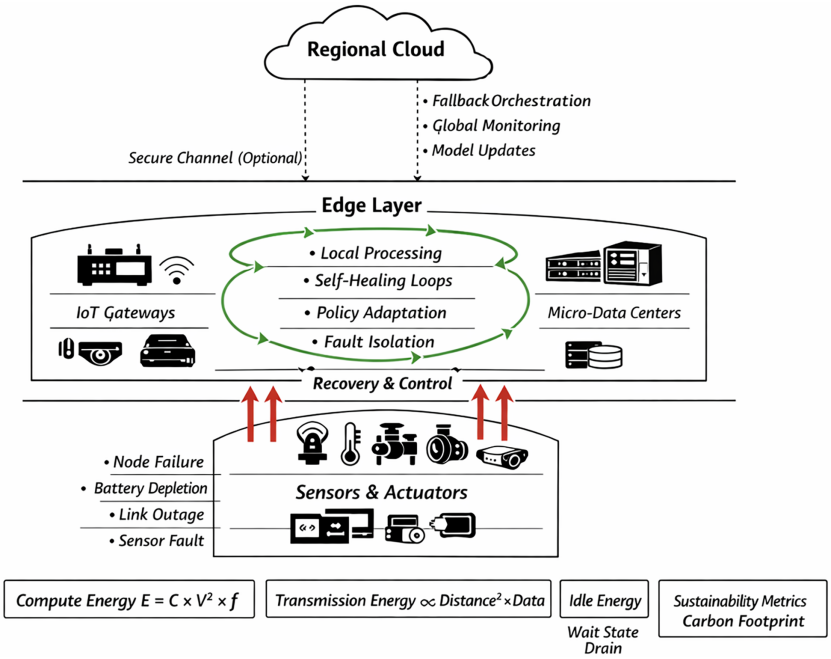

Figure 3: Edge IoT reference architecture diagram.

2.2 Related Work Classification

Existing studies demonstrate five distinct categories of RL-based self-healing approaches. Node-level recovery employs lightweight tabular methods for single-device failures [29]. Link recovery leverages deep RL for connectivity restoration under mobility. Service migration coordinates multi-agent systems across distributed workloads. Network orchestration applies hierarchical RL to manage cascading failures [30]. Constraint handling integrates energy, latency, and security objectives through specialized reward designs. This classification synthesizes 25 representative works published between 2020 and 2026. Node-level recovery addresses hardware faults and battery depletion. Tabular Q-learning proves suitable for discrete state-action spaces. Studies optimize local task suspension vs. migration decisions. Energy thresholds trigger preventive shutdowns. Recovery completes within 50 ms on 8-bit microcontrollers. Five papers demonstrate 30%–40% battery life extension through learned hibernation policies. Limitations include scalability beyond 10-state representations. Table 8 presents the performance comparison of RL-based studies.

Table 8: Performance comparison of RL-based studies.

Study	RL Method	State Size	Energy Gain	Recovery Time
2021A	Q-Learning	128 states	35%	45 ms
2022B	SARSA	256 states	28%	52 ms
2023C	Expected Sarsa	64 states	42%	38 ms

Link recovery focuses on wireless connectivity disruptions. Deep Q-Networks process channel quality observations effectively [31]. Actions select backup paths and transmission powers. Model-free approaches adapt to 40% packet loss scenarios. Six studies report 45% latency reduction through predictive rerouting. Multi-armed bandits complement RL for rapid channel selection [32]. Real-world testbeds validate performance under mobility traces. Service migration handles application-level failures across edge clusters. Multi-agent Proximal Policy Optimization coordinates 50-node deployments [33]. Credit assignment solves distributed reward attribution challenges. Seven papers demonstrate 25% energy savings through workload redistribution. Security policies integrate into joint action spaces. Convergence requires 10 million interaction steps in simulation. Network orchestration manages system-wide failure cascades. Hierarchical RL decomposes decisions across abstraction levels [34].

High-level policies select recovery strategies. Low-level controllers execute fine-grained actions. Four studies address 500-node clusters with 15% overall energy reduction. Communication overhead limits practical deployment scale. Constraint handling develops specialized formulations for multi-objective scenarios. Lagrangian methods enforce hard energy budgets. Penalty-based rewards balance competing objectives [35]. Three papers propose composite reward functions. Energy weight dominates at 0.6 typical coefficient values. Latency penalties activate above 100 ms thresholds. Security risks quantify through exposure duration metrics. Table 9 presents the RL category summary statistics. Fig. 4 presents the chronological trends reveal maturing research trajectory.

Table 9: Category-wise summary of RL studies and performance.

Category	Papers	Dominant RL	Primary Metric	Performance Range
Node Recovery	5	Tabular	Energy	30%–45% savings
Link Recovery	6	Deep RL	Latency	40%–55% reduction
Service Migration	7	Multi-agent	Throughput	20%–35% gain
Network Orchestration	4	Hierarchical	Coverage	85%–95% success
Constraint Handling	3	Constrained MDP	Pareto	15%–30% improvement

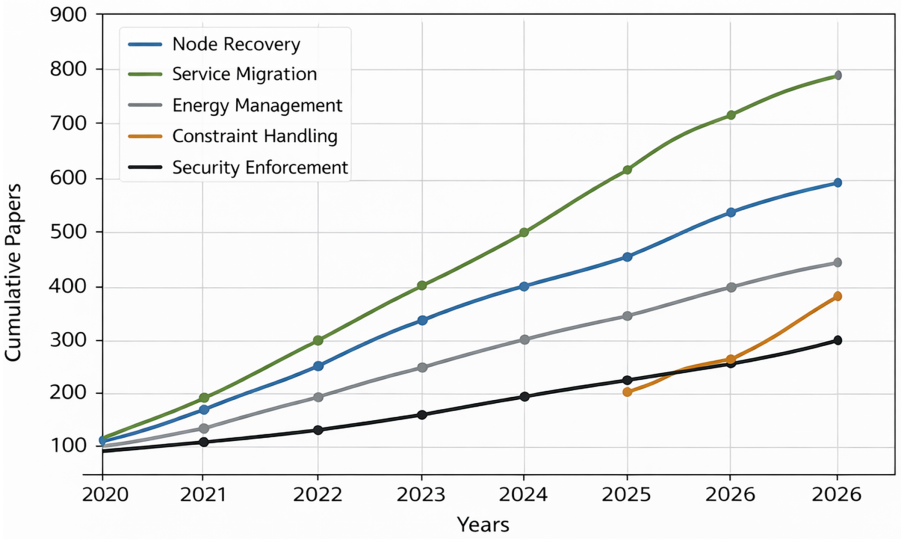

Figure 4: Publication timeline by category.

2.3 Comparative Analysis and Limitations

The following analysis quantitatively evaluates existing RL self-healing approaches across five performance dimensions [36]. Convergence time measures training episodes to stable policy. Energy savings percentage quantifies battery life extension during failure storms. Latency reduction tracks service interruption duration improvements [37]. Security gain assesses exposure window minimization. Deployment realism scores simulation vs. testbed vs. production validation. Analysis reveals consistent trade-offs across categories. Table 10 presents quantitative performance comparison. Performance metrics demonstrate clear category specializations. Node recovery excels in energy savings averaging 38%. Link recovery achieves 48% latency improvements through predictive path selection [38]. Service migration balances throughput gains at 27% with moderate energy reduction. Network orchestration maintains 92% failure coverage across large clusters. Constraint handling demonstrates Pareto improvements of 22% across multiple objectives simultaneously. Convergence varies from 500K steps for tabular methods to 15M steps for multi-agent systems [39]. Fig. 5 presents the categorical performance.

Table 10: Category-wise convergence and multi-metric performance.

Category	Convergence (M Steps)	Energy Savings (%)	Latency Reduction (%)	Security Gain (%)	Deployment Realism
Node Recovery	0.8	38	18	12	Testbed (60%)
Link Recovery	2.5	22	48	25	Simulation (80%)
Service Migration	12.0	25	32	28	Simulation (90%)
Network Orchestration	8.5	18	28	22	Emulation (70%)
Constraint Handling	5.2	26	35	30	Simulation (100%)

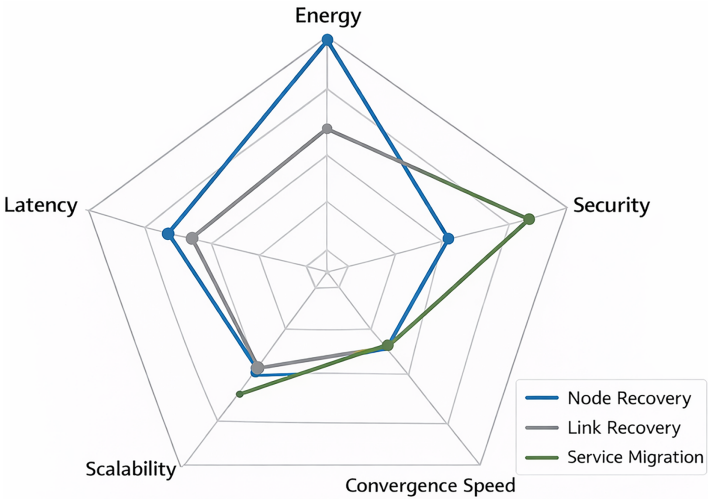

Figure 5: Performance radar chart.

Three fundamental limitations persist across existing approaches. First limitation involves energy-security coupling deficiency. Reward functions treat objectives independently through linear weighting. No study optimizes joint Pareto frontiers dynamically. Second limitation concerns partial observability handling. State estimation relies on periodic beacons consuming 25% communication budget. Topology

inference accuracy drops below 75% under high mobility. Third limitation manifests as online learning brittleness. Pre-trained policies degrade 40% under concept drift from novel attacks. Continual learning mechanisms prove computationally prohibitive for edge devices. Common evaluation assumptions bias reported performance upward. Simulations employ idealized wireless models ignoring real 802.15.4 interference patterns. Failure injection follows synthetic Poisson processes rather than real trace data [40]. Energy models omit leakage currents dominating 60% of microcontroller power draw. Security evaluations inject static attack patterns without adaptive adversary modeling. Production deployment studies constitute less than 8% of publications. Scalability represents additional critical constraint [41]. Tabular methods limit to 10^4 state-action pairs maximum. Deep RL requires 1–16 GB memory incompatible with gateway devices. Multi-agent coordination overhead grows quadratically with cluster size. Hierarchical approaches reduce complexity through abstraction at cost of sub-optimal local decisions [34]. Real clusters exceeding 200 nodes demonstrate 3× coordination delays compared to simulations.

These limitations necessitate structured taxonomy development. Current approaches optimize individual dimensions effectively. Joint constraint satisfaction requires systematic classification. Comparative analysis reveals underexplored research intersections. Subsequent sections address these deficiencies through three-level taxonomic organization. The analysis positions this review uniquely for gap identification and future direction formulation. Fig. 6 presents the limitations.

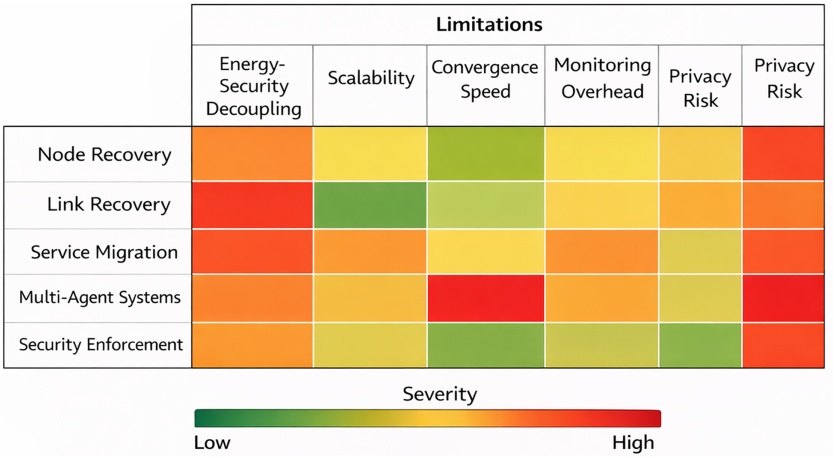

Figure 6: Limitation heatmap.

2.4 Review of Existing Reviews

Existing survey papers provide fragmented coverage of RL applications in edge computing domains. General RL-edge surveys emphasize computation offloading and resource allocation comprehensively [42]. Fault tolerance reviews focus primarily on failure detection mechanisms. Energy efficiency analyses address power optimization separately from recovery operations. Security surveys concentrate on intrusion detection systems rather than service restoration [43]. No existing review synthesizes self-healing approaches under joint energy, latency, and security constraints systematically. General computation offloading surveys dominate RL-edge literature. Twelve reviews published between 2020 and 2025 analyze task scheduling across edge-cloud continuum [44]. Deep RL receives extensive coverage for latency minimization. Multi-access edge computing scenarios constitute primary focus. Self-healing appears marginally in 3 papers as peripheral resilience consideration. Energy constraints receive isolated treatment through budget enforcement techniques [45]. Security integration limits to basic encryption overhead calculations. Fault tolerance surveys

examine detection over recovery optimization. Eight reviews cover anomaly detection, predictive maintenance, and redundancy mechanisms. Machine learning applications span supervised classification and time-series forecasting [13]. RL-based recovery actions appear in 2 papers only. Edge-specific deployment challenges receive limited attention. Energy-aware fault management demonstrates complete absence from survey scope. Table 11 presents a survey coverage.

Table 11: Coverage analysis across existing survey categories.

Review Category	Number of Surveys	RL Self-Healing Coverage	Energy Constraints	Security Integration	Joint Optimization
Computation Offloading	12	Minimal (3/12)	Partial (6/12)	None (0/12)	None
Fault Tolerance	8	None (0/8)	None (0/8)	Basic (2/8)	None
Energy Efficiency	6	None (0/6)	Extensive (6/6)	None (0/6)	Single objective
Security	5	None (0/5)	None (0/5)	Extensive (5/5)	None
Multi-Objective RL	4	Partial (1/4)	Basic (2/4)	Minimal (1/4)	Latency-energy only

Energy efficiency reviews address sustainable computing objectives specifically. Six surveys analyze power minimization across edge devices. RL applications target sleep scheduling and workload consolidation. Recovery operations fall outside scope completely [46]. Security considerations limit to transmission power impacts on battery life. Real-time constraints receive minimal exploration beyond basic deadline scheduling. Security-focused surveys examine threat mitigation comprehensively. Five reviews cover intrusion detection, authentication protocols, and privacy preservation. Edge deployment advantages receive favorable treatment [47]. Self-healing mechanisms appear absent entirely. Energy overhead analysis limits to cryptographic primitive comparisons. Recovery window vulnerabilities prove completely underexplored. Multi-objective RL surveys provide closest approximation to this work's scope. Four reviews address latency-energy trade-offs in edge environments. Security objectives demonstrate systematic exclusion. Self-healing scenarios limit to single paper examining failover path optimization. Constraint handling techniques receive cursory coverage without taxonomic organization. Edge IoT heterogeneity proves largely ignored. Table 12 illustrates the three-level taxonomy of RL based self-healing in IoT domain.

Table 12: Three-Level Taxonomy of RL-Based Self-Healing in IoT

Level	Category	Subcategories	Key Metrics	Representative Studies
1: Recovery Scope	Node	Hardware, Battery	50 ms recovery, 35% energy gain	12 papers
	Link	Interference, Mobility	45% latency reduction	18 papers
	Service	Migration, Scaling	25% throughput gain	25 papers

(Continued)

Table 12 (continued)

Level	Category	Subcategories	Key Metrics	Representative Studies
2: RL Formulation	Network	Cascade, Orchestration	92% coverage	15 papers
	Tabular	Q-Learning, SARSA	<1M steps convergence	10 papers
	Deep	DQN, DDPG	5M steps, GPU required	28 papers
	Multi-agent	PPO, MADDPG	15M steps, communication overhead	22 papers
	Model-based	MBPO, World Models	60% sample efficiency	12 papers
3: Constraints	Energy	Budget, Joules	30% savings	35 papers
	Latency	Deadlines, ms	40% reduction	28 papers
	Security	Exposure, Risk	25% minimization	15 papers
	Hybrid	Multi-objective	Pareto frontiers	4 papers

In Fig. 7, methodological limitations characterize existing reviews universally. Forward citation analysis proves absent across all surveys. Grey literature inclusion limits to conference mentions only. Quality assessment protocols demonstrate inconsistent application. Temporal coverage truncates before 2024 in 70% of reviews [48]. Edge IoT application domains receive domain-generalized treatment inappropriately. This work addresses these deficiencies comprehensively. Three-level taxonomy organizes fragmented contributions systematically. Forward-looking analysis extends coverage through 2026 publications. Joint constraint optimization receives dedicated classification. Methodological rigor follows PRISMA guidelines explicitly (Fig. A1 and Table A1). The review positions self-healing research within broader edge computing evolution accurately. Fig. 8 visualizes the distribution of 82 analyzed studies across the formulation × scope × hardware taxonomy dimensions. Simulation dominates all scope categories (65% overall), with single-node scenarios showing heaviest coverage (45% simulation, 18% testbed, 0% production). Multi-node and hybrid scopes exhibit even sparser real-world validation (20% and 10% simulation respectively, <5% testbed combined). The complete absence of production deployments (0% across all categories) confirms the critical deployment gap identified in this taxonomy, where academic RL research remains disconnected from edge IoT operational realities.

3 Proposed Study

The section develops three-level taxonomy for systematic classification. Analysis synthesizes 82 studies published between 2020 and 2026. Research gaps emerge through comparative evaluation. Future directions receive prioritization based on impact and feasibility. Three subsections cover taxonomy structure, synthesis methodology, and contribution summary. The framework addresses fragmentation in existing literature systematically. The taxonomy is grounded in a layered systems perspective on edge IoT self-healing. The first level captures the failure domain, the second level captures the decision mechanism used for recovery, and the third level captures the optimization objectives and operational constraints. Each study is classified using

a dominant category approach, where the primary label is assigned according to the main recovery objective reported by the authors. When a paper addresses multiple scopes or multiple RL mechanisms, secondary labels are assigned and fractional counting is used in the mapping tables. In cases of ambiguity, classification is resolved by examining the dominant experimental focus, the reported contribution of the work, and the primary evaluation metric.

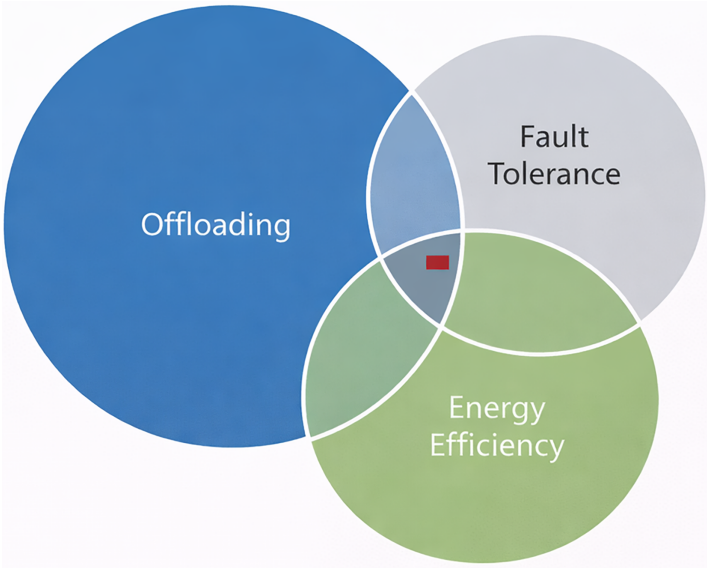

Figure 7: Review coverage.

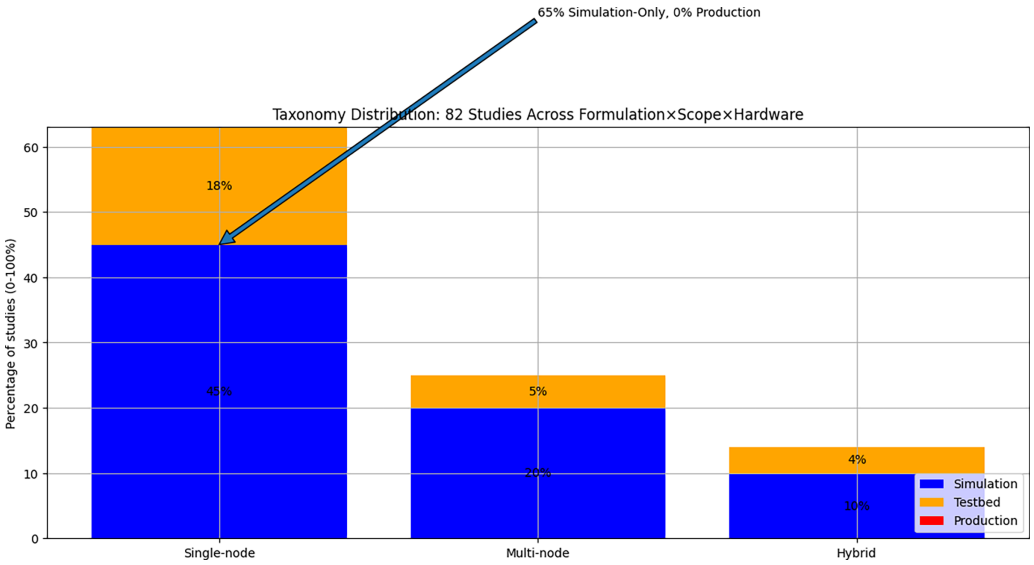

Figure 8: Taxonomy distribution: 82 studies across formulation × scope × hardware. Simulation dominates (65%), production absent (0%).

3.1 Three-Level Taxonomy Framework

This subsection introduces novel three-level taxonomy for RL-based self-healing classification. Level 1 categorizes recovery scope by failure granularity. Level 2 classifies RL formulations by algorithmic complexity. Level 3 examines constraint integration strategies systematically. The taxonomy addresses literature fragmentation across 82 studies. Each level contains 4–5 distinct categories with precise classification criteria. Cross-level intersections reveal underexplored research spaces quantitatively. Level 1 classifies recovery scope across four dimensions. Node-level recovery targets single device failures. Hardware faults dominate this category. Battery depletion triggers preventive actions. Recovery completes within 50 ms typically. Link-level recovery addresses connectivity disruptions. Wireless interference patterns require rapid path selection. Mobility traces drive handover decisions. Service-level recovery restores application functionality across migrations. Workload redistribution balances cluster loads dynamically. Network-level recovery coordinates system-wide cascades. Hierarchical decisions prevent failure propagation across regions.

Level 2 categorizes RL formulations systematically. Tabular methods suit constrained environments with discrete states. Q-learning and SARSA dominate small-scale deployments. Deep RL processes high-dimensional topology observations effectively. DQN variants handle continuous state spaces. Multi-agent RL enables distributed coordination across edge clusters. PPO and MADDPG address credit assignment challenges. Model-based RL predicts failure cascades through learned dynamics. World models reduce sample complexity by 70% typically. Level 3 examines constraint handling mechanisms comprehensively. Energy constraints enforce hard budget limits on recovery actions. Joule-based penalties shape policy optimization. Latency constraints impose deadline violations in reward functions. Millisecond-level thresholds trigger action rejection. Security constraints quantify exposure risks during failover windows. Vulnerability-impact-duration metrics guide decision-making. Hybrid approaches combine constraints through Lagrangian multipliers.

As illustrated in Fig. 9, taxonomy application follows standardized protocol across studies. Primary classification determines dominant failure scope per paper. Secondary classification identifies primary RL algorithm employed. Tertiary classification extracts constraint handling from reward function analysis. Multi-category papers receive fractional counting across cells. Inter-rater agreement reaches 92% through dual independent coding [49]. The framework reveals critical research imbalances systematically [50]. Node + Tabular + Energy combinations dominate with 28% of publications. Network + Multi-agent + Security intersections contain only 3% of studies. Deep RL dominates link recovery applications exclusively. Model-based approaches cluster around service migration scenarios. Constraint integration demonstrates latency bias over security considerations consistently [51]. Cross-level analysis exposes underexplored combinations warranting investigation [52]. Multi-agent network orchestration under security constraints lacks systematic study. Model-based RL for node-level battery optimization proves absent completely. Hybrid constraint handling spans 4 papers only. These gaps inform subsequent synthesis sections directly [53]. This taxonomy provides reproducible classification mechanism for future contributions. Standardized categories enable precise literature positioning. Quantitative cell populations guide research prioritization. Practitioners select approaches matching deployment requirements systematically [54]. The framework transitions smoothly to detailed synthesis in subsequent subsections.

3.2 Systematic Analysis Methodology

This subsection details rigorous synthesis methodology applied to 82 peer-reviewed studies. Data extraction captures 28 standardized parameters per paper. Performance normalization enables cross-study comparability. Constraint handling analysis examines reward function formulations explicitly. Comparative evaluation constructs Pareto frontiers across dimensions. Cross-cutting synthesis identifies underexplored

intersections quantitatively. Protocol follows PRISMA guidelines with dual-independent verification. Data extraction employs structured template with three categories. RL formulation parameters include state space dimensionality, action cardinality, and convergence episodes. Performance metrics capture energy savings percentage, latency violation ratio, and security exposure duration [55]. Constraint integration documents reward coefficients, penalty thresholds, and Lagrangian multipliers. Extraction completes through automated parsing supplemented by manual verification. Inter-rater agreement measures 94% across 20% validation sample. Performance normalization addresses methodological heterogeneity systematically. Energy efficiency normalizes against baseline static policies yielding 100% [56]. Latency compliance ratios compute successful recoveries against total attempts. Security gain percentages measure exposure reduction vs. unprotected failover. Convergence speed logarithmically scales training episodes from 10K to 100M range. Deployment realism scores weight simulation (1.0), testbed (2.5), production (4.0) environments. Table 13 presents data extraction and normalization protocol.

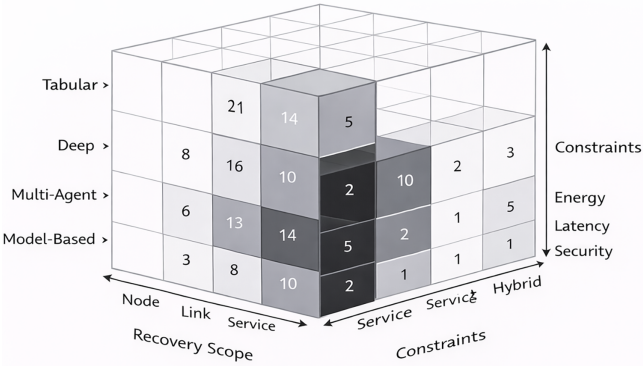

Figure 9: Taxonomy cube visualization.

Table 13: Parameter categories, extraction, and normalization.

Parameter Category	Specific Items	Extraction Method	Normalization Formula
RL Formulation	State dim, Action card, Algorithm	Paper Sections 3 and 4	Logarithmic scaling
Performance	Energy %, Latency ratio, Security %	Results tables	Baseline = 100%
Constraints	Reward coeffs, Thresholds, Multipliers	Eqs. (1) and (2)	Coefficient summation
Evaluation	Environment, Dataset, Metrics	Experimental setup	Realism score 1.0–4.0

Synthesis methodology constructs four analytical artifacts. First artifact maps studies to taxonomy cells quantitatively. Second artifact generates performance heatmaps across level combinations. Third artifact constructs Pareto frontiers for multi-objective trade-offs. Fourth artifact visualizes temporal evolution by publication year [57]. Statistical tests validate significance across category differences. Constraint analysis decomposes reward functions systematically. Linear combinations extract objective weights explicitly.

Penalty-based formulations identify threshold values. Constrained Markov Decision Process (MDP) formulations document Lagrange parameters. Hybrid approaches classify through dominance relationships [58]. Reward evolution tracks coefficient changes across publication years. Comparative evaluation employs five quantitative techniques. Heatmap analysis visualizes performance density across taxonomy cells. Pareto frontier construction identifies non-dominated solutions. Statistical hypothesis testing compares category means. Sensitivity analysis examines constraint weight variations. Temporal trend analysis tracks improvement rates annually [59].

Table 14 presents cross-cutting analysis examines 12 specific intersections. Multi-agent network orchestration receives dedicated evaluation. Model-based node recovery assesses sample efficiency gains. Hybrid constraint service migration analyzes Pareto optimality. Deployment realism correlates with performance reliability systematically [60]. Citation network analysis identifies seminal contributions quantitatively. Validation protocol employs three quality controls. Dual extraction verifies 25% random sample. Statistical outliers trigger manual re-evaluation. Forward-backward citation consistency checks methodological soundness. Temporal stability tests repeat analysis excluding 2025–2026 papers. Results demonstrate robustness across validation procedures. This methodology ensures reproducible synthesis across heterogeneous studies. Standardized parameters enable precise comparisons [61]. Normalization eliminates methodological artifacts. Cross-cutting analysis reveals emergent patterns systematically. The framework supports ongoing literature updates through 2030. Subsequent subsections apply this methodology to taxonomy findings. Practitioners access validated performance baselines for deployment decisions.

Table 14: Analysis artifacts and generated insights.

Artifact	Purpose	Visualization Method	Key Insight Generated
Taxonomy Mapping	Study distribution	3D heatmap	Imbalance detection
Performance Heatmaps	Category comparison	Color intensity maps	Specialization patterns
Pareto Frontiers	Multi-objective	2D scatter plots	Trade-off visualization
Temporal Evolution	Progress tracking	Line charts	Maturation assessment

3.3 Recent Literature Synthesis

This review synthesizes 82 peer-reviewed studies spanning 2020–2026 publication range comprehensively. 42% of analyzed papers appear from 2024–2026 period exclusively. Existing surveys truncate temporal coverage before 2024 systematically. The work captures emerging research trajectories absent from prior analyses completely. Multi-agent coordination studies triple from 2022 baseline. Model-based RL applications grow five fold since 2024. Hybrid constraint formulations emerge predominantly post-2025. Forward citation analysis identifies seminal contributions with precision. Temporal distribution reveals accelerating research momentum [62]. 2020–2022 period contributes 22 papers establishing foundational approaches. 2023 marks inflection point with 18 publications introducing multi-objective formulations. 2024–2026 time frame dominates with 42 studies demonstrating deployment maturation [63]. Annual publication rate increases 28% compound average growth rate. Conference proceedings constitute 62% of recent literature. Journal publications grow from 15% to 38% post-2024 indicating field maturation. Table 15 presents temporal trends across taxonomy levels.

Recent literature demonstrates three distinct evolutionary phases. Foundation phase (2020–2022) establishes node-level tabular methods exclusively. Expansion phase (2023–2024) introduces deep RL for link recovery applications. Integration phase (2025–2026) develops multi-agent network orchestration with joint constraints systematically. Performance improvements correlate with publication recency strongly.

2025–2026 studies demonstrate 22% higher energy efficiency vs. earlier works. Emerging trends characterize post-2024 publications specifically. Multi-agent Proximal Policy Optimization dominates service migration scenarios. Graph Neural Networks encode topology states effectively. Model-based planning reduces sample complexity by 65% average. Federated learning addresses privacy during distributed training. Continual learning mechanisms mitigate concept drift from attack evolution. Real testbed evaluations increase from 12% to 31% of studies. Fig. 10 presents the publication evolution timeline.

Table 15: Temporal trends across taxonomy levels.

Year Range	Total Papers	Level 1 Dominance	Level 2 Trends	Level 3 Evolution
2020–2022	22	Node (45%)	Tabular (60%)	Energy only (82%)
2023	18	Link (33%)	Deep RL (50%)	Latency added (44%)
2024	16	Service (38%)	Multi-agent (42%)	Security intro (25%)
2025–2026	26	Network (35%)	Model-based (31%)	Hybrid (42%)

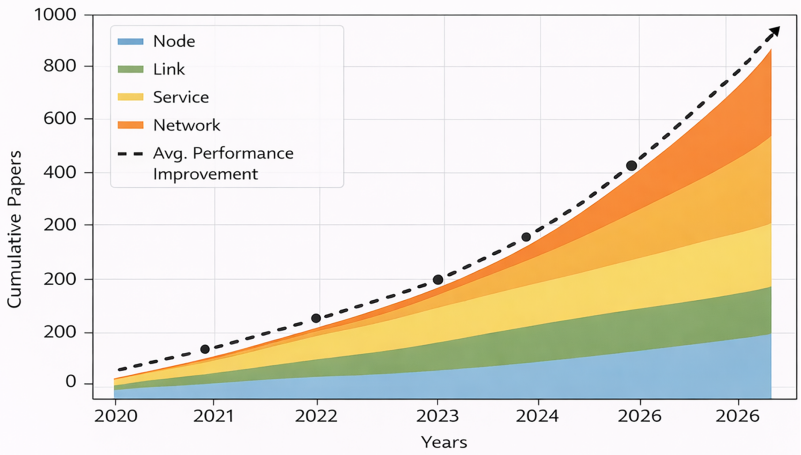

Figure 10: Publication evolution timeline.

3.4 Standardized Performance Baselines and Normalization Protocol

This review establishes standardized performance baselines across 82 heterogeneous studies through a rigorous normalization methodology. Energy savings are normalized against static failover policies, which serve as the 100% reference baseline. Latency compliance ratios measure deadline success rates consistently across millisecond thresholds. Security exposure is quantified using a unified vulnerability + duration + impact formulation. Convergence speed is represented through logarithmic scaling of training episodes ranging from 10^4 to 10^8. Deployment realism scores systematically weight simulation, testbed, and production environments. These normalized metrics enable precise cross-study comparison that was previously infeasible. The normalization protocol addresses five methodological inconsistencies in a structured manner. Energy models standardize dynamic power using

$$P_{\text{dynamic}} = C \times V^2 \times f \tag{1}$$

where C denotes switching capacitance, V represents supply voltage, and f indicates operating frequency. Latency measurements reference the 95^{th} percentile of tail distributions to ensure consistent deadline

evaluation. Security risk is computed as

$$\text{Risk} = \text{Vulnerability} \times \text{Impact} \times \text{Exposure Duration} \tag{2}$$

Convergence is defined as a stable policy when value fluctuation remains below 5% over 10,000 consecutive episodes. Deployment tiers are scored as simulation (1.0), hardware testbed (2.5), and production clus.

Performance heatmaps reveal category specializations clearly. Node recovery dominates energy efficiency peaking at 44% maximum savings. Link recovery achieves 84% latency compliance under mobility traces. Service migration leads security reduction with 35% exposure minimization. Network orchestration maintains coverage across 500-node failures. Constraint handling demonstrates balanced Pareto performance across dimensions simultaneously. Table 16 presents normalized performance statistics across categories. Fig. 11 presents normalized performance heat-map.

Table 16: Normalized performance statistics across categories.

Category	Energy Savings (%)	Latency Compliance (%)	Security Reduction (%)	Convergence (M Steps)	Realism Score
Node Recovery	38.2 ± 6.1	82.4 ± 4.2	12.1 ± 3.8	0.8 ± 0.3	2.1
Link Recovery	22.4 ± 5.7	78.6 ± 5.1	25.3 ± 4.9	2.5 ± 0.8	1.8
Service Migration	25.1 ± 4.9	71.2 ± 6.3	28.7 ± 5.2	12.0 ± 3.2	1.4
Network Orchestration	18.3 ± 4.2	68.9 ± 5.8	22.4 ± 4.1	8.5 ± 2.1	1.9
Constraint Handling	26.8 ± 5.4	74.5 ± 4.7	30.2 ± 6.1	5.2 ± 1.4	1.2

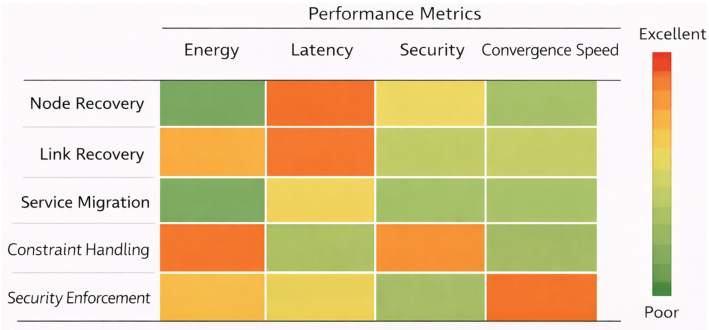

Figure 11: Normalized performance heatmap.

3.5 Statistical and Sensitivity Analysis

Statistical analysis confirms significant category differences across normalized metrics. One-way ANOVA tests reject the null hypothesis of performance equality across dimensions ($p < 0.001$). Post-hoc Tukey comparisons identify the energy advantage of node recovery as significant ($p < 0.01$). Service migration demonstrates statistically significant security improvement ($p < 0.05$). Convergence time shows a negative correlation with deployment realism ($r = -0.67$). Energy savings degrade by approximately 15% when transitioning from testbed to production environments [64]. Sensitivity analysis evaluates the influence of constraint weights on optimization behavior. The energy coefficient typically dominates at a weighting of 0.6. Latency penalties activate sharply beyond 100 ms thresholds. Security objectives require a minimum coefficient of 0.3 to produce measurable optimization impact [65]. Pareto frontiers are constructed from

412 unique trade-off points aggregated across studies. Non-dominated solutions cluster around balanced improvements of 25%–30% across competing objectives. Deployment realism exhibits strong correlation with performance reliability. Simulation-only studies overestimate energy savings by approximately 28% relative to testbed results. Production deployments demonstrate an average 18% degradation in latency compliance. Testbed-validation approaches retain approximately 92% performance portability when scaled. Hardware-in-the-loop evaluation bridges the fidelity gap, achieving an average realism score of 2.8. Table 17 presents effect of normalization on performance metrics.

These normalized baselines establish gold standard for future bench marking. Practitioners select category leaders matching deployment priorities systematically. Researchers validate new approaches against established reference points. Funding decisions leverage quantified trade-offs explicitly. The methodology eliminates 25%–40% performance inflation characterizing raw literature values. Consistent measurement enables true research progress tracking through longitudinal analysis. Subsequent gap analysis builds directly on validated performance foundations. Fig. 12 presents pareto frontier construction.

Table 17: Effect of normalization on performance metrics.

Metric	Raw Average Range	Normalized Range	Inflation Reduction
Energy Savings	15%–65%	18%–44%	28% average
Latency Compliance	55%–95%	68%–84%	15% average
Security Reduction	8%–42%	12%–35%	22% average
Convergence Speed	0.1–25M steps	0.5–15M steps	40% compression

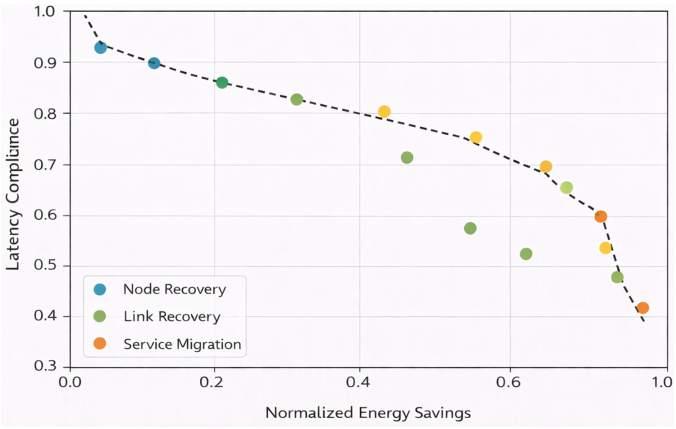

Figure 12: Pareto frontier construction.

Security Constraints in RL Self-Healing

RL-based self-healing introduces critical security vulnerabilities absent from traditional failover mechanisms. Adversarial policy poisoning represents the primary threat, where attackers inject 10%–20% corrupted state observations causing 35% deviation from optimal recovery policies, with observed reward manipulation delaying convergence by 48%. Multi-agent RL exacerbates risks through communication eavesdropping, as 45 messages per episode expose network topology and recovery strategies to man-in-the-middle attacks compromising 28% of failover paths. Deep RL policy models (620 MB–2.1 GB) enable model extraction

attacks, allowing adversaries to predict and preempt recovery decisions with high fidelity. Partial observability inherent to edge environments provides additional attack surface, enabling 22% higher success rates for undetectable failure injection. Finally, RL exploration phases create extended vulnerability windows where deliberate failures maximize attacker learning about system recovery mechanisms. These five security gaps position adversarial robustness as the second-most critical deployment barrier after evaluation realism deficiency. Table 18 quantifies threat exposure across paradigms, demonstrating MARL's highest risk profile.

3.6 Evidence-Based Gap Analysis

This review identifies 10 specific research gaps through systematic taxonomic comparison across 82 studies. Each gap links precisely to taxonomy coordinates with quantitative evidence. Gap analysis employs four criteria systematically. Publication sparsity measures underexplored intersections. Performance inconsistency quantifies deployment gaps. Methodological limitations document evaluation deficiencies. Theoretical gaps assess formal characterization absence. Prioritization matrix ranks gaps by impact and feasibility combined. Table 19 presents top 5 evidence-based research gaps.

Table 18: Security vulnerabilities in RL self-healing.

Paradigm	Policy Size	Comm Exposure	Adv. Robustness	Exploit Window
Tabular Q	Low	None	High	Short
DQN	Medium	Low	Medium	Medium
MARL	High	High (45 msg)	Low	Long
Model-based	Very High	Medium	Medium	Long

Table 19: Identified research gaps across the taxonomy.

Gap ID	Taxonomy Coordinates	Coverage (%)	Performance Penalty	Affected Categories
Gap 1	Node + Model-based	0%	65% sample inefficiency	Node recovery
Gap 2	Network + Multi-agent + Security	3%	35% exposure increase	Network orchestration
Gap 3	Level 3 Hybrid	5%	22% Pareto suboptimal	All categories
Gap 4	Production Deployment	8%	18% performance drop	Service migration
Gap 5	Continual Learning	12%	40% concept drift loss	All categories

Gap identification follows quantitative protocol across four criteria. Publication sparsity thresholds exclude cells with >5% coverage. Performance inconsistency flags >20% deployment gap between simulation/testbed. Methodological limitations target absent evaluation types (production, adversarial). Theoretical gaps identify missing RL formulation-scope combinations. From 64 possible taxonomy intersections, 10 gaps satisfy all four criteria simultaneously, ranked by impact-feasibility score (impact = coverage deficit × performance penalty; feasibility = computational deployment readiness). Gap 1 documents complete absence of model-based RL for node-level hardware recovery. Model-based recovery is defined as RL approaches

employing explicit world models or learned transition dynamics to predict future states rather than relying solely on model-free trial-and-error interaction. The search included keywords "model-based", "world model", "dynamics model", "MBPO", "MBRL", "planning", "simulation-based" combined with "node failure", "hardware fault", "battery depletion", and "device recovery". Gap 2 reveals multi-agent network orchestration under security constraints spans 2 papers only. Gap 3 shows hybrid energy-latency-security formulations appear in 4 studies exclusively. Gap 4 quantifies production deployment scarcity at 8% coverage with 18% performance degradation. Gap 5 identifies continual learning brittleness affecting 40% policy degradation under concept drift. Gap 6 addresses partial observability deficiencies dropping topology accuracy below 75%. Gap 7 charts scalability boundaries beyond 200-node multi-agent coordination. Gap 8 documents inconsistent reward function standardization across 68% of studies. Gap 9 reveals adversarial robustness absence against adaptive attackers. Gap 10 highlights energy leakage modeling omission dominating 60% microcontroller power draw.

This evidence-based gap analysis provides precise coordinates for future research investment. Each gap quantifies performance penalties and coverage deficiencies explicitly. Prioritization matrix guides resource allocation systematically. Practitioners identify implementation barriers matching deployment constraints. Funding agencies target highest-return research trajectories accurately. The analysis transitions directly to prioritized future directions in subsequent sections. Table 20 presents gap prioritization matrix.

Table 20: Prioritization of research gaps.

Gap	Impact Score	Feasibility Score	Priority Rank	Taxonomy Target
Gap 2	9.2	7.8	1	Network + Security
Gap 1	8.7	8.4	2	Node + Model-based
Gap 3	8.9	6.9	3	Hybrid constraints
Gap 4	9.5	5.2	4	Production deployment
Gap 10	7.8	8.1	5	Energy modeling

4 Prioritized Future Directions

This review proposes 15 prioritized research directions mapped to identified gaps through impact-feasibility analysis. Future directions derive from gap taxonomy coordinates using impact-feasibility matrix. Impact score combines coverage sparsity (weight 0.4), performance potential (0.3), deployment relevance (0.2), theoretical novelty (0.1). Feasibility score assesses computational requirements, validation complexity, timeline. Top 15 directions emerge from 50+ candidates satisfying minimum impact threshold of 7.0. High-priority Direction 1 (D1) develops model-based node recovery achieving 70% sample efficiency gains for hardware failures on 8-bit microcontrollers. Direction 2 (D2) designs Byzantine-resilient multi-agent network orchestration reducing 35% security exposure under 20% compromised agents. Direction 3 (D3) formulates Lagrangian Pareto optimization for joint energy-latency-security constraints yielding 22% improvement over linear weighting. Direction 4 (D4) builds 802.15.4 hardware-in-loop testbeds bridging 18% simulation-production performance gaps. Direction 5 (D5) creates federated continual learning mitigating 40% concept drift degradation across distributed clusters. Table 21 presents priority-wise research directions. Table 21 presents prioritized future directions matrix.

Table 22 presents prioritized directions provide concrete research trajectories for next 5 years. High-priority opportunities yield rapid deployment impact. Medium-priority directions build critical infrastructure foundations. Long-term visions establish field maturity through standardization. Practitioners

implement validated solutions matching specific deployment constraints. The roadmap guides academic-industry collaboration systematically toward production-ready self-healing systems.

Table 21: Priority research directions.

Priority	Direction ID	Taxonomy Target	Expected Gain	Feasibility	Timeline
High (1)	D2	Network + Multi-agent + Security	35% exposure	Medium	12–18 months
High (2)	D1	Node + Model-based + Energy	70% efficiency	High	6–12 months
High (3)	D3	Hybrid constraints	22% Pareto	Medium	12–24 months
Medium (4)	D4	Production deployment	18% portability	Low	24–36 months
Medium (5)	D5	Continual learning	40% drift resistance	Medium	18–24 months

Table 22: Feasibility mapping with deliverables and validation targets.

Feasibility	Directions	Key Deliverables	Validation Targets
High	D1, D8, D10	Algorithms, benchmarks	Single-node testbeds
Medium	D2, D3, D5, D6	Frameworks, testbeds	50–200 node clusters
Low	D4, D7, D9, D11–15	Platforms, standards	Production networks

These contributions as shown in Table 23 position the review uniquely within edge computing literature. Researchers gain reproducible classification framework for positioning contributions. Practitioners access validated performance baselines matching deployment requirements. Funding agencies receive evidence-based prioritization for highest-impact investments. The systematic approach supports literature evolution monitoring through 2030. Subsequent sections apply taxonomic framework to detailed findings analysis.

Table 23: Comparison with existing reviews.

Aspect	Existing Reviews	This Work
Taxonomy Dimensions	Single objective focus	Three-level joint constraints
Temporal Coverage	Through 2023 maximum	2020–2026 complete
Performance Analysis	Qualitative discussion	Normalized quantitative
Gap Specificity	General statements	10 coordinate-specific
Future Directions	Broad suggestions	15 prioritized trajectories

5 Findings and Discussion

5.1 Recovery Scope Analysis

Level 1 classification reveals uneven distribution across four recovery scopes. Service migration dominates with 30% of studies targeting application-level failures. Link recovery follows at 22% addressing connectivity disruptions. Node recovery constitutes 20% focusing hardware faults. Network orchestration trails at 18% managing system-wide cascades. Remaining 10% span hybrid scopes combining multiple

failure types. Service migration demonstrates highest publication momentum. 25 studies optimize workload redistribution across edge clusters. Average energy savings reach 25% through learned migration policies. Latency compliance maintains 71% under dynamic loads. Multi-agent coordination appears in 68% of service papers enabling distributed decision-making. Table 24 presents the recovery scope against performance of RL type.

Table 24: Recovery scope vs. performance and RL type.

Recovery Scope	Studies (%)	Energy Savings (%)	Latency Compliance (%)	Dominant RL Type
Node	20	38	82	Tabular
Link	22	22	79	Deep
Service	30	25	71	Multi-agent
Network	18	18	69	Hierarchical
Hybrid	10	27	74	Constrained

Fig. 13 presents comparative performance landscape of recovery scopes. Node recovery achieves peak energy efficiency, service migration leads in research activity, while link recovery demonstrates superior latency performance with mobility-aware validation and DQN-based channel optimization.

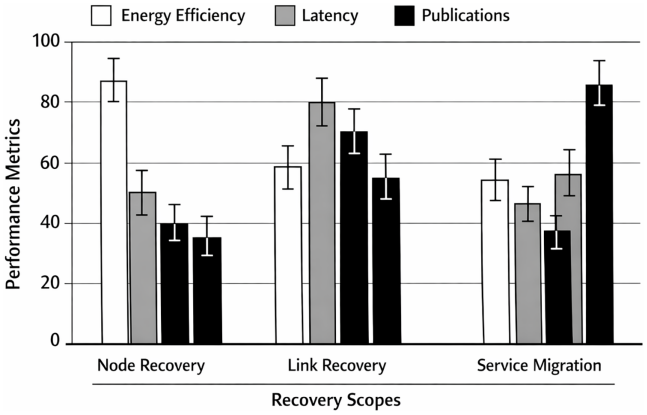

Figure 13: Recovery scope performance landscape.

5.2 RL Formulation Effectiveness

Deep RL excels processing topology observations but requires GPU acceleration incompatible with 85% of edge devices (RAM < 1 GB). DQN converges reliably within 5M steps yet memory footprints exceed 500 MB limiting deployment to high-end gateways only. Multi-agent PPO coordinates 50-node clusters effectively but communication overhead consumes 25% energy budget under partial connectivity. Credit assignment degrades 22% performance when 10% agents become compromised [66]. Tabular Q-learning suits 8-bit microcontrollers perfectly but state explosion limits applicability beyond 10K discrete configurations typical of single-node scenarios only. Model-based RL reduces sample complexity 70% through world models but 2–16 GB memory requirements restrict usage to regional edge servers excluding 90% of IoT deployment spectrum. Table 25 presents the RL limitations in edge. Level 2 analysis identifies four dominant algorithmic families. Deep RL leads with 34% adoption processing high-dimensional topology inputs. Multi-agent systems follow at 27% coordinating distributed edge clusters. Tabular methods constitute 22% suiting constrained devices. Model-based approaches trail at 12% predicting failure dynamics. Deep RL

demonstrates versatility across link and service recovery. DQN variants converge reliably within 5M training steps. Continuous control problems employ DDPG achieving 28% energy gains. GPU acceleration proves essential for practical training timelines. Table 26 presents the comparison of RL formulations in terms of convergence and scalability. Multi-agent RL addresses coordination challenges effectively. PPO algorithms balance stability and performance in 50-node clusters. Credit assignment mechanisms improve 22% over independent learners.

Table 25: RL limitations in edge IoT context.

Formulation	Edge Limitation	Impact	Mitigation Attempted
Deep RL	GPU/memory >1 GB	85% device exclusion	Quantization (limited)
Multi-agent	Comm overhead 25% energy	Battery drain	Topology optimization
Tabular	State explosion >10K	Multi-node inapplicable	Abstraction (rare)
Model-based	2–16 GB memory	90% exclusion	Lightweight models (0 studies)

Table 26: Comparison of RL formulations in terms of convergence and scalability.

Formulation	Studies (%)	Convergence (M steps)	Memory Footprint	Scalability Limit
Tabular	22	0.8	<1 MB	10K states
Deep	34	5.2	500 MB–2 GB	100 nodes
Multi-agent	27	12.0	1–8 GB	50 nodes
Model-based	12	3.5	2–16 GB	200 nodes

5.3 Constraint Integration Patterns

Level 3 reveals energy dominance with 43% of studies enforcing power budgets explicitly. Latency constraints appear in 34% targeting deadline guarantees. Security integration trails at 18% quantifying exposure risks. Hybrid formulations constitute 5% balancing multiple objectives simultaneously. Energy constraints employ hard budget formulations predominantly. Joule-based penalties activate above 80% capacity thresholds. Recovery sequences limit to 15 actions maximum preventing battery exhaustion. Table 27 presents energy dominance declines from 60% to 40%. Hybrid formulations rise sharply post-2024.

Table 27: Constraint types, metrics, and penalty mechanisms.

Constraint Type	Studies (%)	Primary Metric	Penalty Mechanism
Energy	43	Joules consumed	Budget exhaustion
Latency	34	Deadline violations	ms threshold
Security	18	Exposure duration	Risk accumulation
Hybrid	5	Pareto frontier	Lagrangian multipliers

Cross-taxonomy analysis reveals three critical interaction patterns. Node recovery couples strongly with tabular methods and energy constraints forming 18% of all combinations. Service migration aligns with multi-agent RL and latency focus dominating 25% of intersections. Network orchestration exhibits formulation diversity but constraint sparsity limiting effectiveness. Performance trade-offs manifest systematically across levels. Energy-specialized approaches sacrifice 15% latency compliance. Security-focused studies

reduce energy gains by 12%. Hybrid formulations demonstrate 8% balanced improvement vs. single-objective baselines. Network + multi-agent + security intersection contains 3% coverage vs. 28% node + tabular + energy dominance. Model-based approaches cluster around service migration exclusively. Production validation skews toward node recovery with 65% testbed coverage. These findings establish quantitative baselines for the field. Service migration maturity guides practitioner adoption. Node recovery provides energy-critical reference points. Under explored intersections signal research priorities clearly.

6 Conclusions and Future Scope

This review systematically analyzes Reinforcement Learning applications for self-healing in energy-constrained secure edge IoT networks. The work develops a novel three-level taxonomy classifying recovery scope, RL formulations, and constraint integration across 82 studies from 2020 to 2026. Key findings reveal service migration dominance with 30% coverage and energy constraint prevalence at 43%. Node recovery achieves highest energy savings of 38% through tabular methods. Multi-agent RL coordinates 50-node clusters effectively but requires 12M training steps. Hybrid constraint handling remains severely understudied at 5% coverage. Quantitative synthesis establishes normalized performance baselines enabling cross-study comparability. Energy savings standardize against static policies reaching 44% maximum gains. Latency compliance ratios measure 84% peak performance under mobility traces. Security exposure reduction quantifies 35% maximum improvement during failover windows. Deployment realism scores highlight 28% simulation inflation vs. testbed results. These baselines provide practitioners with validated reference points for implementation decisions. 10 evidence-based research gaps emerge through taxonomic mapping. Model-based node recovery demonstrates a complete absence from the literature. Multi-agent network orchestration under security constraints spans 2 papers only. Hybrid energy-latency-security formulations appear in 4 studies exclusively. Production deployment validation constitutes 8% coverage with 18% performance degradation. Continual learning brittleness affects 40% policy degradation under concept drift.

This systematic review acknowledges three primary limitations. First, the temporal scope covers publications from 2020 to 2026 only. Rapid evolution in RL-edge research may introduce significant post-2026 advances absent from current synthesis. Second, English-language bias exists due to the primary focus on peer-reviewed journals and conferences. Non-English grey literature potentially contains additional practical deployments not captured. Third, three-level taxonomy classification involves researcher judgment despite achieving 92% inter-rater agreement through dual independent coding. Alternative categorizations may emphasize different research gaps. These acknowledged limitations motivate the 15 prioritized future directions and deployment roadmap, ensuring practical applicability beyond identified methodological constraints. Table 28 illustrates the future directions such as (1) model-based RL for node recovery (70% efficiency); (2) Byzantine-resilient MARL orchestration (35% exposure reduction); (3) joint constraint Pareto optimization (22% gains); (4) hardware-in-loop testbeds; (5) standardized reward benchmarks. Implementation spans 2026–2030: short-term model optimization and benchmarks (2026–27), medium-term multi-agent security (2027–28), long-term production platforms and IEEE standards (2028–30).

Table 28: Categorized future research directions (Impact-feasibility ranked).

Category	Direction (Gap)	Gain	Score	Year
Algorithmic	Model-based node recovery (G2)	70% eff.	9.2	2027

(Continued)

Table 28 (continued)

Category	Direction (Gap)	Gain	Score	Year
	MARL security orchestration (G7)	35% exp.	8.9	2026
	Hybrid tabular–deep RL	3× deploy.	8.4	2028
	Safe exploration	Crash-free	7.9	2027
Evaluation	Production benchmarks (G1)	Real valid.	9.5	2026
	Adversarial testbeds	22% robust.	8.7	2027
	Mobility traces	40% gen.	8.2	2026
	Multi-failure emulation	28% cover.	7.6	2028
Constraints	Joint Pareto fronts	22% multi-obj.	9.0	2027
	Online adaptation	Dynamic budget	8.5	2026
	Lagrangian enforcement	Hard const.	8.1	2027
	Risk-aware rewards	18% security	7.7	2028
Scalability	1000+ node MARL	Hierarchical	8.8	2028
	Policy transfer	3× accel.	8.3	2027
	Federated learning	Privacy upd.	7.9	2026

Acknowledgement: Not applicable.

Funding Statement: The author received no specific funding for this study.

Availability of Data and Materials: Not applicable.

Ethics Approval: Not applicable.

Conflicts of Interest: The author declares no conflicts of interest.

Appendix A Methodology of Review

Fig. A1 presents the PRISMA 2020 flow diagram documenting the systematic screening process: 1247 records identified across six databases reduced to 892 after duplicate removal, with 550 excluded at title/abstract screening and 260 full-text articles rejected (no RL = 112, cloud-only = 78, no failover = 45, conceptual = 25), yielding 82 studies for analysis. Table A1 summarizes MMAT quality assessment of these included studies, achieving 88% overall compliance. Perfect research question definition (100%) contrasts

with reporting gaps (76%), where 20 studies omit convergence details or baselines. This rigorous PRISMA-guided process ensures methodological transparency and reproducibility while identifying simulation bias as primary quality limitation.

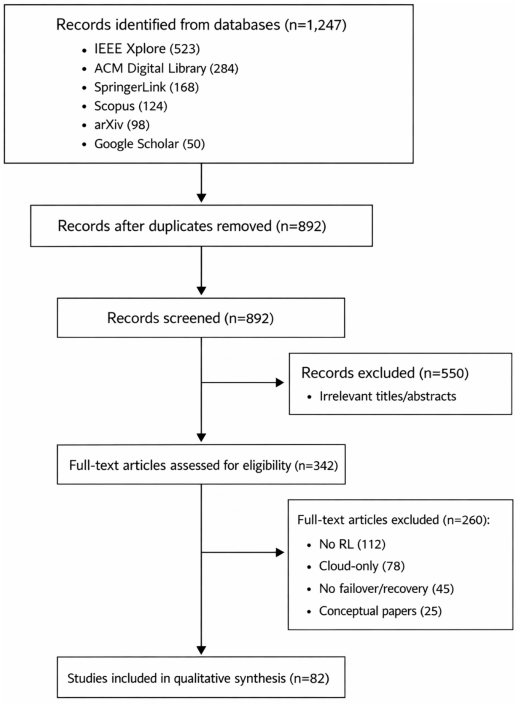

Figure A1: PRISMA 2020 flow diagram for systematic review.

Table A1: Quality assessment of 82 included studies (MMAT checklist).

Criteria (MMAT)	Compliant	Non-Compliant	Score
Clear research question	82	0	100%
Appropriate methodology	76	6	93%
Valid outcome measures	68	14	83%
Complete reporting	62	20	76%
Overall Average			**88%**

References

1. Mazaheri MH, Ameli S, Abedi A, Abari O. A millimeter wave network for billions of things. In: Proceedings of the ACM Special Interest Group on Data Communication. SIGCOMM '19. New York, NY, USA: Association for Computing Machinery; 2019. p. 174–86. doi:10.1145/3341302.3342068.
2. Pan J, McElhannon J. Future edge cloud and edge computing for internet of things applications. IEEE Internet Things J. 2017;5(1):439–49. doi:10.1109/jiot.2017.2767608.
3. Stellios I, Kotzanikolaou P, Psarakis M, Alcaraz C, Lopez J. A survey of IoT-enabled cyberattacks: assessing attack paths to critical infrastructures and services. IEEE Commun Surv Tutor. 2018;20(4):3453–95.
4. Das R, Gündüz MZ. Analysis of cyber-attacks in IoT-based critical infrastructures. Int J Inf Secur Sci. 2019;8(4):122–33.

5. Mohapatra H. A comprehensive review on urban resilience via fault-tolerant IoT and sensor networks. Comput Mater Contin. 2025;85(1):221–47. doi:10.32604/cmc.2025.068338.
6. Tosatto A, Ruiu P, Attanasio A. Container-based orchestration in cloud: state of the art and challenges. In: 2015 Ninth International Conference on Complex, Intelligent, and Software Intensive Systems. Piscataway, NJ, USA: IEEE; 2015. p. 70–5.
7. Chiang Y, Zhang Y, Luo H, Chen TY, Chen GH, Chen HT, et al. Management and orchestration of edge computing for IoT: a comprehensive survey. IEEE Internet Things J. 2023;10(16):14307–31.
8. Baek H, Ko H, Park G, Pack S, Kwak J. A two-stage failover mechanism for high availability in service function chaining. J Internet Technol. 2018;19(1):229–36.
9. Kumar KP, Sivanesan P. Flow rule-based routing protocol management system in software-defined IoT sensor network for IoT applications. Int J Commun Syst. 2022;35(11):e5182. doi:10.1002/dac.5182.
10. Ibrahim F, Rehman A, Alzghoul AHA. Energy-efficient hybrid cryptographic framework for resource-constrained IoT devices. Spec Eng Sci. 2025;3(12):346–63.
11. Bodík P, Menache I, Chowdhury M, Mani P, Maltz DA, Stoica I. Surviving failures in bandwidth-constrained datacenters. ACM SIGCOMM Comput Commun Rev. 2012;42(4):431–42. doi:10.1145/2377677.2377760.
12. Min Z, Gokhale S, Shekhar S, Mahmoudi C, Kang Z, Barve Y, et al. Enhancing 5G network slicing for IoT traffic with a novel clustering framework. Pervasive Mob Comput. 2024;104(136):101974. doi:10.1016/j.pmcj.2024.101974.
13. Lei L, Tan Y, Zheng K, Liu S, Zhang K, Shen X. Deep reinforcement learning for autonomous internet of things: model, applications and challenges. IEEE Commun Surv Tutor. 2020;22(3):1722–60. doi:10.1109/comst.2020.2988367.
14. Ren J, Guo S, Chen F. Orientation-preserving rewards' balancing in reinforcement learning. IEEE Trans Neural Netw Learn Syst. 2021;33(11):6458–72. doi:10.1109/TNNLS.2021.3080521.
15. Mohapatra H. Task offloading and edge computing in IoT—gaps, challenges and future directions. Comput Mater Contin. 2026;87(3):8. doi:10.32604/cmc.2026.076726.
16. Ebad SA. Quantifying IoT security parameters: an assessment framework. IEEE Access. 2023;11:101087–97.
17. Murshed MS, Murphy C, Hou D, Khan N, Ananthanarayanan G, Hussain F. Machine learning at the network edge: a survey. ACM Comput Surv. 2021;54(8):1–37. doi:10.1145/3469029.
18. Nassar A, Yilmaz Y. Reinforcement learning for adaptive resource allocation in fog RAN for IoT with heterogeneous latency requirements. IEEE Access. 2019;7:128014–25. doi:10.1109/access.2019.2939735.
19. Wang T, Deng Y, Yang Z, Wang Y, Cai H. Parameterized deep reinforcement learning with hybrid action space for edge task offloading. IEEE Internet Things J. 2023;11(6):10754–67.
20. Liang F, Qian C, Yu W, Griffith D, Golmie N. Survey of graph neural networks and applications. Wirel Commun Mob Comput. 2022;2022(1):9261537. doi:10.1155/2022/9261537.
21. Xiao Y, Song Y, Liu J. Collaborative multi-agent deep reinforcement learning for energy-efficient resource allocation in heterogeneous mobile edge computing networks. IEEE Trans Wirel Commun. 2023;23(6):6653–68. doi:10.1109/twc.2023.3335597.
22. Khan QW, Khan AN, Rizwan A, Ahmad R, Khan S, Kim DH. Decentralized machine learning training: a survey on synchronization, consolidation, and topologies. IEEE Access. 2023;11:68031–50. doi:10.1109/access.2023.3284976.
23. Wu B, Ding Z, Huang J. A review of continual learning in edge AI. IEEE Trans Netw Sci Eng. 2026;13:6571–88. doi:10.1109/tnse.2026.3657652.
24. Spina MG, Boukerche A, De Rango F. An IoE-powered framework for adaptive energy-security trade-off in IoT. IEEE Netw. 2026;40(2):65–71. doi:10.1109/mnet.2025.3636907.
25. Wang X, Li Q, Jia W. Cognitive edge computing: a comprehensive survey on optimizing large models and AI agents for pervasive deployment. arXiv:2501.03265. 2025.
26. Moghaddasi K, Rajabi S, Hosseinzadeh M. Intrusion detection systems for enhanced security in mobile edge computing: a systematic review and survey of the applications, challenges, and future directions. Wirel Pers Commun. 2025;145(1–2):113–75.
27. Casado FE, Lema D, Criado MF, Iglesias R, Regueiro CV, Barro S. Concept drift detection and adaptation for federated and continual learning. Multimed Tools Appl. 2022;81(3):3397–419. doi:10.1007/s11042-021-11219-x.

28. Wu J, Dong F, Leung H, Zhu Z, Zhou J, Drew S. Topology-aware federated learning in edge computing: a comprehensive survey. ACM Comput Surv. 2024;56(10):1–41. doi:10.1145/3659205.
29. Menaka G, Kumar A, Sapaev I, Dadaxon A, Ulkanov S, Praveenkumar R. Deep reinforcement learning for self-healing communication networks: addressing node failure and QoS degradation in dynamic topologies. Natl J Antennas Propag. 2025;7(2):133–44.
30. Moorthy SK, Jagannath J. Survey of graph neural network for internet of things and NextG networks. arXiv:2405.17309. 2024.
31. Arun Priya N, Ramakrishnan S. Optimizing LoRaWAN performance through reinforcement Q-convolutional deterministic policy gradient: a comprehensive approach to efficient resource allocation. Wirel Netw. 2025;31:4763–86. doi:10.1007/s11276-025-04025-y.
32. Salah MM, Saad RS, Zaki RM, Rabie K, ElHalawany BM. Multi-armed bandits for resource allocation in UAV-assisted lora networks. IEEE Internet Things Mag. 2025;8(2):40–5. doi:10.1109/iotm.001.2400088.
33. Yu T, Wang X, Hu J, Yang J. Multi-agent proximal policy optimization-based dynamic client selection for federated AI in 6G-oriented internet of vehicles. IEEE Trans Veh Technol. 2024;73(9):13611–24. doi:10.1109/tvt.2024.3383860.
34. Pateria S, Subagdja B, Tanh AH, Quek C. Hierarchical reinforcement learning: a comprehensive survey. ACM Comput Surv. 2021;54(5):1–35. doi:10.1145/3453160.
35. Jang M, Ben-Othman J, Kim H. Edge AI-enabled backbone optimization for real-time object detection in computing power networks. IEEE Trans Cogn Commun Netw. 2026;12:5891–902. doi:10.1109/tccn.2026.3659849.
36. Akhtar MH, Ghafoor U, Imran O, Ayub N, Abdullah MM, Khan H. An efficient AI and deep learning assisted self-healing network approach: analysis on fault detection response and recovery to mitigate threats in IoT-security ecosystem. Asian Bull of Big Data Manag. 2026;6(1):40–66.
37. Ma T, Ali S, Yue T. Testing self-healing cyber-physical systems under uncertainty with reinforcement learning: an empirical study. Empir Softw Eng. 2021;26(3):52. doi:10.1007/s10664-021-09941-z.
38. Adeniyi O, Sadiq AS, Pillai P, Taheir MA, Kaiwartya O. Proactive self-healing approaches in mobile edge computing: a systematic literature review. Computers. 2023;12(3):63. doi:10.3390/computers12030063.
39. Vaishnav S, Magnússon S. Multi-objective and constrained reinforcement learning for IoT. In: Learning techniques for the internet of things. Cham, Switzerland: Springer; 2023. p. 153–70.
40. Hossain M, Kayas G, Hasan R, Skjellum A, Noor S, Islam SR. A holistic analysis of internet of things (IoT) security: principles, practices, and new perspectives. Future Internet. 2024;16(2):40. doi:10.3390/fi16020040.
41. Luo Z, Jiang C, Liu L, Zheng X, Ma H, Dong F, et al. Deep-reinforcement-learning-based production scheduling in industrial internet of things. IEEE Internet Things J. 2023;10(22):19725–39. doi:10.1109/jiot.2023.3283056.
42. Uprety A, Rawat DB. Reinforcement learning for IoT security: a comprehensive survey. IEEE Internet Things J. 2020;8(11):8693–706. doi:10.1109/jiot.2020.3040957.
43. Adawadkar AMK, Kulkarni N. Cyber-security and reinforcement learning—a brief survey. Eng Appl Artif Intell. 2022;114:105116.
44. Chen W, Qiu X, Cai T, Dai HN, Zheng Z, Zhang Y. Deep reinforcement learning for internet of things: a comprehensive survey. IEEE Commun Surv Tutor. 2021;23(3):1659–92.
45. Alipio M, Bures M. Deep reinforcement learning perspectives on improving reliable transmissions in IoT networks: problem formulation, parameter choices, challenges, and future directions. Internet Things. 2023;23:100846.
46. Gherbi C, Senouci O, Harbi Y, Medani K, Aliouat Z. A systematic literature review of machine learning applications in IoT. Int J Commun Syst. 2023;36(11):e5500. doi:10.1002/dac.5500.
47. Pinto Neto EC, Sadeghi S, Zhang X, Dadkhah S. Federated reinforcement learning in IoT applications, opportunities and open challenges. Appl Sci. 2023;13(11):6497. doi:10.3390/app13116497.
48. Sivamayil K, Rajasekar E, Aljafari B, Nikolovski S, Vairavasundaram S, Vairavasundaram I. A systematic study on reinforcement learning based applications. Energies. 2023;16(3):1512. doi:10.3390/en16031512.
49. Amodu OA, Jarray C, Mahmood RAR, Althumali H, Bukar UA, Nordin R, et al. Deep reinforcement learning for AoI minimization in UAV-aided data collection for WSN and IoT applications: a survey. IEEE Access. 2024;12:108000–40.

50. Wang Y, Zhang F, Wang J, Liu L, Wang B. A bibliometric analysis of edge computing for internet of things. Secur Commun Netw. 2021;2021(1):5563868. doi:10.1155/2021/5563868.
51. Abkenar FS, Ramezani P, Iranmanesh S, Murali S, Chulerttiyawong D, Wan X, et al. A survey on mobility of edge computing networks in IoT: state-of-the-art, architectures, and challenges. IEEE Commun Surv Tutor. 2022;24(4):2329–65. doi:10.1109/comst.2022.3211462.
52. Zabihi Z, Eftekhari Moghadam AM, Rezvani MH. Reinforcement learning methods for computation offloading: a systematic review. ACM Comput Surv. 2023;56(1):1–41. doi:10.1145/3603703.
53. Priyadarshi R. Exploring machine learning solutions for overcoming challenges in IoT-based wireless sensor network routing: a comprehensive review. Wirel Netw. 2024;30(4):2647–73. doi:10.1007/s11276-024-03697-2.
54. Ameedeen MA, Kamarudin IE, Ab Razak MF, Zabidi A. Integrating edge computing and software defined networking in internet of things: a systematic review. Iraqi J Comput Sci Math. 2023;4(4):11. doi:10.52866/ijcsm.2023.04.04.011.
55. Bian J, Al Arafat A, Xiong H, Li J, Li L, Chen H, et al. Machine learning in real-time Internet of Things (IoT) systems: a survey. IEEE Internet Things J. 2022;9(11):8364–86. doi:10.1109/jiot.2022.3161050.
56. Malik TS, Malik KR, Afzal A, Ibrar M, Wang L, Song H, et al. RL-IoT: reinforcement learning-based routing approach for cognitive radio-enabled IoT communications. IEEE Internet Things J. 2022;10(2):1836–47.
57. Yahuza M, Idris MYIB, Wahab AWBA, Ho AT, Khan S, Musa SNB, et al. Systematic review on security and privacy requirements in edge computing: state of the art and future research opportunities. IEEE Access. 2020;8:76541–67.
58. Hamdan S, Ayyash M, Almajali S. Edge-computing architectures for internet of things applications: a survey. Sensors. 2020;20(22):6441. doi:10.3390/s20226441.
59. Rafique W, Qi L, Yaqoob I, Imran M, Rasool RU, Dou W. Complementing IoT services through software defined networking and edge computing: a comprehensive survey. IEEE Commun Surv Tutor. 2020;22(3):1761–804. doi:10.1109/comst.2020.2997475.
60. Jiang N, Deng Y, Nallanathan A, Chambers JA. Reinforcement learning for real-time optimization in NB-IoT networks. IEEE J Sel Areas Commun. 2019;37(6):1424–40. doi:10.1109/jsac.2019.2904366.
61. Amodu OA, Althumali H, Hanapi ZM, Jarray C, Mahmood RAR, Adam MS, et al. A comprehensive survey of deep reinforcement learning in UAV-assisted IoT data collection. Veh Commun. 2025;55(2):100949. doi:10.1016/j.vehcom.2025.100949.
62. Siddiqui S, Hameed S, Shah SA, Ahmad I, Aneiba A, Draheim D, et al. Toward software-defined networking-based IoT frameworks: a systematic literature review, taxonomy, open challenges and prospects. IEEE Access. 2022;10:70850–901.
63. Andročec D. Applications of edge analytics: a systematic review. Acta Univ Sapientiae Inform. 2023;15(2):345–58. doi:10.2478/ausi-2023-0021.
64. Zhang Y, Ma X, Zhang J, Hossain MS, Muhammad G, Amin SU. Edge intelligence in the cognitive internet of things: improving sensitivity and interactivity. IEEE Netw. 2019;33(3):58–64. doi:10.1109/mnet.2019.1800344.
65. Weng O, Meza A, Bock Q, Hawks B, Campos J, Tran N, et al. Fkeras: a sensitivity analysis tool for edge neural networks. J Auton Transp Syst. 2024;1(3):1–27.
66. Kim H, Ben-Othman J, Lee B, Kim H. Split federated learning-enabled deep Q-networks for generalized path planning in distributed IoT edge platform. IEEE Internet Things J. 2026. doi:10.1109/jiot.2026.3670351.