

Design and Evaluation of Self-Healing Networks Using Q-Learning and Digital Twin Technology

L. S. Abdul Kadhim 

Department of Computer and Network Engineering, Hakim Sabzevari University, Iran

Article information

Article history:

Received: March 29, 2026

Revised: May 24, 2026

Accepted: June 16, 2026

Available online: October 01, 2026

Keywords:

Self-Healing Networks

Reinforcement Learning

Digital Twin

Network Fault Management

Autonomous Systems

Q-Learning

Correspondence:

Laith Sami Abdul Kadhim

laithsami2024@gmail.com

Abstract

The expanding complexity of communication networks nowadays requires reliability, resilience, and quick fault recovery as must-have features for future communication infrastructures. Current fault management solutions are largely manual-based or rule-based static systems, and cannot adequately deal with dynamic and massive network faults. Thus, the purpose of this research is to design and test an intelligent self-healing network framework based on the integration of Reinforcement Learning (RL) and the Digital Twin technology for autonomous, adaptive, and real-time self-healing. The main idea of the proposed system is to design an optimal repair strategy learning agent based on Q-learning that continuously interacts with a simulated network environment and to continuously simulate the physical network with the Digital Twin to predict potential failures according to the historical data and real-time data of the network. The architecture proposed consists of several processing modules, including a network simulator that supports various network topologies including small-world, random, adaptive, scale-free, complete, star and grid networks, a node and edge condition type monitoring and prediction module based on Digital Twins, an ϵ -greedy Q learning algorithm for making adaptive decisions in the repair process, and a composite health score mechanism based on node and edge conditions. The system also utilizes failure prediction models, historical data management, and reward optimization techniques for enhancing learning convergence and network resilience. The criteria used for the evaluation are: network health, mean time to repair (MTTR), break detection accuracy, false positive rate, network availability, and repair success. Experimental results show that the proposed RL-Digital Twin framework significantly outperforms the traditional repair method of doing nothing, random repair, and a rule-based repair method, with an average network health of 89.4%, an average repair time of 3.1 steps, and a failure prediction accuracy of 91.2%. The RL agent can adapt to various network topologies and converge in 65–100 training episodes. Despite these benefits, the study has been restricted to simulated networks up to 30 nodes, and mainly considers operational failures as opposed to cybersecurity attacks. In addition, scalability challenges remain for larger networks due to the limitations of traditional Q-learning methods.

DOI: [10.33899/jes.v35i4.62252](https://doi.org/10.33899/jes.v35i4.62252), ©Authors, 2026, College of Education for Pure Science, University of Mosul.

This is an open access article under the CC BY 4.0 license (<http://creativecommons.org/licenses/by/4.0/>).

1. Introduction

The lightning speed of the development of communication networks has resulted in the more and more complicated and dynamic infrastructures, in which such performance measures as reliability, resilience, and adaptability are of paramount importance. The attributes of modern network architectures (5G, O-RAN, and 6G) are heterogeneous devices, multi-layered protocols, and large-scale distributed components. As much as these networks provide unparalleled connectivity and

serviceability, they are also complex and therefore more likely to experience faults, fail, and have a decline in performance [1]. Conventional network management methods that have traditionally been based on human observation and inflexible settings are not always sufficient to uphold an optimum functioning in the dynamic environment or heavy-scale failures. This has led to the development of self-healing networks, which can automatically identify, diagnose, and remedy faults, becoming a key focus of research in order to provide high network availability and service continuity [2].

Self-healing networks are networks that rely on smart decision-making resources in order to track network well-being, identify irregularities, and implement remedial measures automatically. Initial solutions were either rule-based systems, threshold monitoring, or a fixed redundancy solution [3]. Although these approaches may have been useful to deal with foreseeable faults and local failures, they were not always scalable, adaptable, and capable of dealing with sophisticated cascading failures through large network structures. Moreover, the repair mechanisms that are not made dynamic may add unwanted overhead since they might not be able to dynamically schedule critical nodes and optimal repair sequences based on the dynamic conditions of the network [4].

The application of Artificial Intelligence (AI) to network management has boosted self-healing functionality a lot. Fault detection, anomaly classification, and predictive maintenance have been attempted using machine learning (ML) methods, such as supervised and unsupervised learning techniques. Decision trees, support vector machines, and neural networks are supervised learning techniques that can be used to accurately classify network events using historical data. Nevertheless, these models can also assume labeled datasets and can be ineffective at extrapolating to faults that have not been observed before or new network conditions. There are methods of unsupervised learning, such as clustering and anomaly detection algorithms, which do not have these limitations since they do not need labels to detect the deviations in behavior. Although they are effective in identifying anomalies, these methods have a low capacity to make sequential and long-term repair decisions, which are essential in reducing downtimes and ensuring the overall health of the network [5].

Because of its capacity to learn optimal policies of making sequential decisions in response to the environment, Reinforcement Learning (RL) has become a promising self-healing network candidate. With RL, an agent monitors the state of the network, acts (e.g., repairing nodes, rerouting traffic), and is rewarded in terms of network performance. With time, the agent gets to optimize cumulative rewards, which essentially would give preference to critical repairs, cascading failures minimization, and the dynamism of network situations. Q-Learning is a popular model-free RL method where agents can estimate action-state pair value without knowledge of network dynamics and, therefore, is suitable for complex, partially observable network models. With the inclusion of RL in self-healing mechanisms, networks will be able to implement adaptive and proactive repair mechanisms that enhance as more operational information is made available [6].

A vital instrument to enable RL-based self-healing networks is Digital Twin (DT) technology that offers a high-fidelity virtual replica of the physical network. Digital twins recreate the network topology, device configuration, traffic patterns, and failure behavior, enabling the RL agent to act as if it were a real environment and test the interventions it can do therein, without risk. Such a virtual feedback loop speeds up the learning process, minimizes the chances of real-world interruption, and allows a systematic testing of the corrective measures in various failure conditions. Also, digital twins support predictive maintenance and predict possible failures and determine corrective actions, which mitigate the impact on live traffic, improving reliability and decreasing Mean Time To Repair (MTTR) [7].

The recent research has shown that RL and digital twin technology combined can be used to heal a network. As a case study, agents based on RL have been demonstrated to optimize repair paths, keep networks healthy, and adapt to various topologies, such as Random Geometric, Scale-Free, and Small-World networks. These strategies enhance the repair success rate as well as minimize the delay in repair and the use of resources. Nevertheless, there are still some problems, such as properly representing the state of the network in the digital twin, managing large-scale networks with changing traffic patterns, and combining multi-objective measures, such as reliability, latency, and cost, into the RL reward function [8].

Besides optimization of performance, security, and fault tolerance have become significant concerns for the self-healing network. The automatic repair systems need to differentiate between legitimate failures and malicious attacks, like Denial-of-Service (DoS) or ransom attacks of network control nodes. A combination of AI-based anomaly detection and RL-based repair mechanisms can help networks react dynamically to unintentional failure and adversarial events, increasing resilience and robustness. Moreover, adaptive learning models can continuously enhance repair policies based on dynamic network states, traffic patterns, and new attack patterns, and it fits the vision of intelligent and autonomous networks in 6G and beyond [4].

This study does not primarily work on cybersecurity defense, although modern communication networks are becoming more vulnerable to security attacks like Denial-of-Service (DoS) and malicious attacks, infrastructure-level reliability and fault recovery mechanisms are the focus of this study. In particular, this paper focuses on non-malicious network failures such as node failures, edge failures, traffic overloads, and cascading failures that are common in large-scale and dynamic network settings. The proposed Reinforcement Learning and Digital Twin-based framework aims to boost the network's resilience, self-healing ability, and stability in normal and stress situations by concentrating on these operational failures. This study could be extended to incorporate security-aware learning mechanisms to handle adversarial scenarios, but it is not in the scope of this study.

Current communication systems are highly complicated and susceptible to various faults, including node, link faults, and cascading faults. The current rule-based and manual approaches to managing these networks are not only inefficient but also incapable of offering real-time self-recovery services. The aforementioned challenges indicate that there is a requirement for developing an intelligent and automated fault management mechanism. The primary goal of this research is to design a fully automated self-healing network through the application of Reinforcement Learning (RL) and Digital Twins. The proposed system will be capable of learning an optimal approach to repairs, predicting network faults ahead of time, and restoring communication networks in real time. Specifically, the project will target improvement in terms of network performance, recovery time, accuracy in failure detection, and network availability.

1.1 Problem Statement

Due to the quick rise in IoT applications, cloud computing, 5G/6G technologies, and intelligent infrastructure, communication networks have advanced and grown more dynamic, massive, and complicated. The current methods of network administration and self-repair rely on manual management or rule-based techniques, making them less adaptable to changing environments. Some of the shortcomings associated with these approaches include delayed detection of faults, high costs related to recovery, poor scalability, inability to detect potential network failure, lack of predictive capabilities, and inefficiency in heterogeneous environments. Most of the current self-healing approaches tend to concentrate on either fault detection or reinforcement learning without incorporating the use of predictive digital twins to facilitate proactive maintenance and autonomous self-healing. With failures having a significant impact on network performance, availability, and downtime, it is crucial to develop an approach for self-healing that would be able to monitor and predict network faults as well as make autonomous decisions. For this reason, the current research aims at solving the problem of designing a scalable and efficient self-healing communication network utilizing Q-learning and Digital Twin technologies.

1.2 Research Objectives

This paper aims to design and evaluate an intelligent self-healing network framework using Reinforcement Learning (RL) and Digital Twin (DT) technology for autonomous fault detection, prediction, and recovery in communication networks. The particular goals are:

1. To create a Digital Twin model that can be continuously synced with the real communication network and track network conditions in real time.
2. To create a reinforcement learning agent that can learn optimal repair strategies by interacting with dynamic network environments using Q-learning.
3. To develop proactive failure prediction mechanisms based on the historical data of the network and the sliding window analysis.
4. To generate a holistic Network Health Score, based on Node Condition, Edge Performance, Bandwidth, Latency, and Failure Status.
5. To test the proposed framework across various network topologies such as random, scale-free, small world, complete, star, and grid networks.
6. To increase network resilience by reducing recovery time, minimising false alarms, increasing repair success rate, and enhancing network availability.
7. To compare performance between the proposed RL + Digital Twin framework and the traditional approaches of No-Healing, Random Repair, Rule Based repair system.
8. To study the scalability, convergence behavior, and deployment aspects of the proposed self-healing architecture for modern communication networks.

1.3 Research Contributions

In this paper, we propose an innovative self-healing framework for networks using a combination of Reinforcement Learning (Q-Learning) and Digital Twin technology. Previous works only focused on utilizing Digital Twins either for predicting and monitoring the status of the network or using reinforcement learning for making decisions. However, our proposed model uses both approaches simultaneously to monitor the real-time network status, predict future faults, and take appropriate actions to recover from those faults without human intervention. Digital Twins are constantly updating the current status of the physical environment while using past experiences and current data to predict future possibilities. Our proposed RL agents then analyze this information and take suitable actions based on that. Apart from these unique contributions, our model also uses cascading failures with a combination of edge and node-based health scores, multiple topologies, and priority-based repair techniques. The main novelty and contributions:

1. Integration of reinforcement learning and digital twin into one unified self-healing scheme for the communication network.
2. Moving from reactive fault tolerance towards proactive and predictive fault tolerance via failure prediction.
3. Utilizing information from Digital Twin to enhance learning/prediction precision with its historical data analysis and sliding window approach.
4. Development of a unique metric for assessing the state of the network based on both node and edge parameters.

5. Adaptability to various network architectures, including scale-free, small-world, star, grid, complete, and random networks.
6. Cascading failures detection and targeted restoration of important network components by prioritizing them.
7. Decrease in the Mean Time To Repair (MTTR) with increased network availability/stability.
8. Improvement in the quality of decision-making by decreasing the rate of unsafe exploration of possible actions.

2. Related Works

The recent progress of wireless communication networks into an open and intelligence-oriented design has attracted interest to the field of wireless communication technology, and in particular with Open Radio Access Networks (O-RAN) and the recently about-to-launch 6G networks. O-RAN is a design that tries to make the conventional radio access network design open, and the network is introduced with intelligence to be managed and optimized by incorporating artificial intelligence and machine learning features in the network system design. Nevertheless, this higher complexity level of O-RAN and 6G networks has enhanced the risks in the network design regarding the security capabilities and performance of the network functionality without burdening energy efficiency, since many researchers have targeted the technical parameters of the new technology.

Almasan et al. [9] present the notion of Network Digital Twins (NDTs), which represents an important enabler technology for future communication networks. Digital twins are characterized as real-time digital representations of a physical network that constantly synchronize with actual network data. The paper describes how digital twins assist in monitoring the network, detecting faults, analyzing its performance, and conducting preventive maintenance. Through virtualizing network behavior in a simulated environment, operators can perform failure simulation and devise recovery tactics without impacting actual operations. It is emphasized that incorporating machine learning within digital twins increases their prediction capability and decision-making ability. Practical applications encompass resource management, traffic optimization, and failure prediction in 5G and IoT domains.

Mismar and Evans [10] focus on the application of reinforcement learning for self-healing management of failures within a communication network. A self-healing framework based on reinforcement learning has been proposed, wherein the reinforcement learning agent learns the best policy or repair actions through interaction with the simulated network environment. The reinforcement learning algorithm applies feedback learning for detecting, isolating the faulty nodes, and correcting the error without any human intervention. In contrast to conventional approaches, the proposed reinforcement learning algorithm dynamically adjusts to any changes in the network and any unknown patterns of failure. The goal of reinforcement learning is to maximize the stability of the network while minimizing the downtime. Experimental results have demonstrated that reinforcement learning offers efficient fault recovery time and network stability under different scenarios of network failure. Although the application of reinforcement learning becomes more challenging for larger networks, its efficacy has been validated in smaller simulation environments.

Saxena et al. [11] provide a model of self-healing and self-recovery digital twin for cloud computing environments. The authors examine the issue of enhancing the reliability of distributed systems by providing an effective way to deal with possible disruptions. It should be noted that the suggested architecture involves the combination of predictive analytics and automatic recovery capabilities for the purpose of ensuring the stability of the system. One of the crucial innovations introduced into the system is the application of machine learning techniques to predict failures and allocate resources accordingly. At the same time, collaborative digital twin synchronization allows one to maintain synchronization of the virtual and real environment.

Cheng et al. [12] studied digital twin utilization with RL algorithms to optimize resource utilization in the network and help in recovery after any network faults. It is noted that there is convergence inefficiency and risky exploration involved in traditional reinforcement learning in the physical network. The use of the digital twin environment helps to address both these issues by serving as a virtual testbed for training an RL algorithm. This enables the RL model to explore the repair process without actually failing the real network. The result of this research includes faster convergence rates, accurate decisions, and improved stability. The case studies reveal promising results in terms of lower latency, efficient fault recovery, and better resource utilization.

Majid and Isreal [13] reviewed various methodologies for automating communication networks using AI with an emphasis on self-healing and self-organizing networks. The authors describe the current demands for intelligent techniques to detect, analyze, and recover from network issues within a communication system without any human assistance. This study also traces the development from using rule-based fault management techniques to intelligent systems relying on machine learning and reinforcement learning methods. Self-healing networks are referred to as those networks that adapt to faults automatically by employing recovery plans and optimizing network topologies and resources. The importance of reinforcement learning in improving recovery solutions in terms of performance by interacting with their environment is discussed, along with the use of predictive models to detect possible failures earlier.

Hakiri et al. [14] focused on digital twin networks (DTNs) as one of the most important innovations in order to manage network operation and troubleshoot failures. Digital twins provide real-time virtual models of real-life communication networks that can be constantly monitored and tested. One of the most significant aspects of DTN is highlighted as the opportunity for operators to analyze different failure situations as well as test their recovery options in a virtual environment. This research offers an

insight into the use of machine learning techniques, like reinforcement learning, in order to optimize the decision-making process and network operation. One of the main benefits described by the authors is increased fault detection rates and reduced mean time to repair.

Ward et al. [15] were concerned with artificial intelligence methods for detecting and recovering from faults in self-healing communication networks. It demonstrates that conventional approaches used in managing network operations lack efficiency in processing extensive faults. To address this issue, the research proposes the use of intelligent algorithms, including machine learning and reinforcement learning, for detecting anomalies, identifying faults, and performing automated recovery operations. Self-healing processes continuously analyze network conditions and learn from the outcomes of prior operations to enhance future decisions. The study proves that the reinforcement learning technique can be utilized for enhancing decision-making through learning optimal repair policies. This approach relies less on pre-defined rules. Therefore, the findings reveal considerable advancements in fault tolerance, minimized network interruptions, and accelerated fault recovery.

Noun et al. [16] have reviewed the use of reinforcement learning techniques in fault diagnosis, and tolerant control systems have been reviewed. The paper points out that RL algorithms are able to learn an optimal policy through their interactions with the environment. They are applied in network fault management in order to detect any faults, make appropriate decisions, and modify their strategy depending on the feedback obtained from the system. RL techniques are classified into model-free, model-based, and deep reinforcement learning systems. The advantages of RL in terms of flexibility, scalability, and optimal performance in uncertain environments are pointed out. However, some issues associated with convergence, lack of interpretability, and safety of exploration processes are highlighted.

Dinter et al. [17] focused on the significance of digital twin technology for maintenance management in complex and networked systems. According to this work, digital twins create a continuous real-time replica of the physical systems that allows the prediction of failure modes. This systematic review describes how historical and current data, along with simulation models, are used in the context of digital twins to inform decision-making on maintenance. In communication networks, the application of digital twins allows identifying and addressing faults ahead of time. Some difficulties associated with applying digital twins, namely the lack of historical information, heavy calculations, and high-level complexity of the models, have also been identified.

Yang et al. [18] examined the use of reinforcement learning (RL) in self-management, particularly in the application of the approach to network self-healing and self-adaptation in communication systems. The researchers note that existing fault management systems utilize pre-defined rules, which cannot work effectively in large and complex networks due to their inefficiency. Reinforcement learning becomes a potential alternative since, in this method, the agent learns how to make decisions regarding system repair and configuration by interacting with the environment of the network. Through state space, actions, and reward definitions, the agent learns the right behavior for ensuring minimal downtime in the network. It is noted that, in contrast to other methods, RL allows one to continuously adapt to the changes in the network and new failure cases.

Omar et al. [19] provided an analysis of different aspects related to self-healing technologies in next-generation communication networks, including those of the 5G type and beyond. In the beginning, self-healing is defined as the capability of a network to automatically identify, analyze, and recover from any possible failure without any need for human interaction. The research paper identifies three main types of self-healing, namely, reactive, proactive, and predictive. While the former responds to already occurred problems, the latter two use data analytics to predict potential issues to take timely actions to avoid failures. The paper pays much attention to the significance of artificial intelligence methods, especially machine learning and reinforcement learning, in the process of making decisions.

Kauffman et al. [20] examined the impact of digital twin technology on the development of intelligent and autonomous communication networks. Digital twins are described as virtual replicas of a physical network that imitate its functioning, status, and performance. The authors emphasize the capabilities of digital twins in detecting faults, monitoring performance, and predicting maintenance. Through constant synchronization with the real-time information from the physical network, digital twins make it possible to simulate faults and evaluate solutions without disrupting active processes. Furthermore, this paper explains how the use of artificial intelligence and machine learning technologies contributes to the improvement of prediction and decision-making.

Mujezinović et al. [21] explore methods for fault detection and restoration in communication networks with the help of artificial intelligence techniques. We state that the growing complexity of networks implies the need for sophisticated techniques able to detect and recover from faults independently. We conduct a review of existing machine learning and reinforcement learning approaches employed in fault detection, anomaly detection, and fault recovery. In particular, we emphasize the efficiency of reinforcement learning algorithms as a way to learn effective fault recovery policies through interactions with the network. We provide examples of using hybrid approaches incorporating both prediction and decision-making models in order to enhance the robustness of systems.

Madala et al. [22] examined the use of reinforcement learning algorithms in network optimization problems such as fault management, routing, and resource management. In this research, the network is represented as a Markov Decision Problem whereby the RL agents learn how to make optimal decisions by interacting with the environment. The objective of the reward

function used is to maximize network efficiency while minimizing network failures. It has been demonstrated that the RL algorithms perform better than conventional optimization methods under dynamic and uncertain network environments. The benefits of using RL algorithms include adaptability, scalability, and autonomous decision-making. However, training time is identified as a challenge in the application of RL.

3. Methodology

The approach presents a new model of autonomous self-healing networks by combining Reinforcement Learning (RL) and Digital Twin technology. The strategy targets the shortcomings of the traditional network management systems that, in most cases, operate either on a manual basis or through a rule-based system that is not dynamic. The proposed system, in its turn, allows making intelligent and real-time decisions and proactively managing faults. Reinforcement Learning is used to enable an intelligent agent to learn the best repair strategies by means of constant interaction with the network environment, whereas the Digital Twin keeps a virtual version of the physical network, which can be monitored, predicted, and analyzed. The framework works in a closed-loop fashion whereby the system monitors the network condition, identifies failures, pre-determines the right repair measures, and modifies its learning policy through feedback. The reward mechanism is formulated so as to harmonize various goals, which comprise maximization of network health, minimization of failures, and issuing rewards to successful repairs. Another feature of the Digital Twin is that it amplifies the intelligence of the system by using past experiences with the view to anticipating possible failures in order to initiate reactive and proactive remedial measures. The system architecture, as shown in Figure 1, is built to include four primary elements: the Network Simulator, Digital Twin, RL Agent, and User Interface.

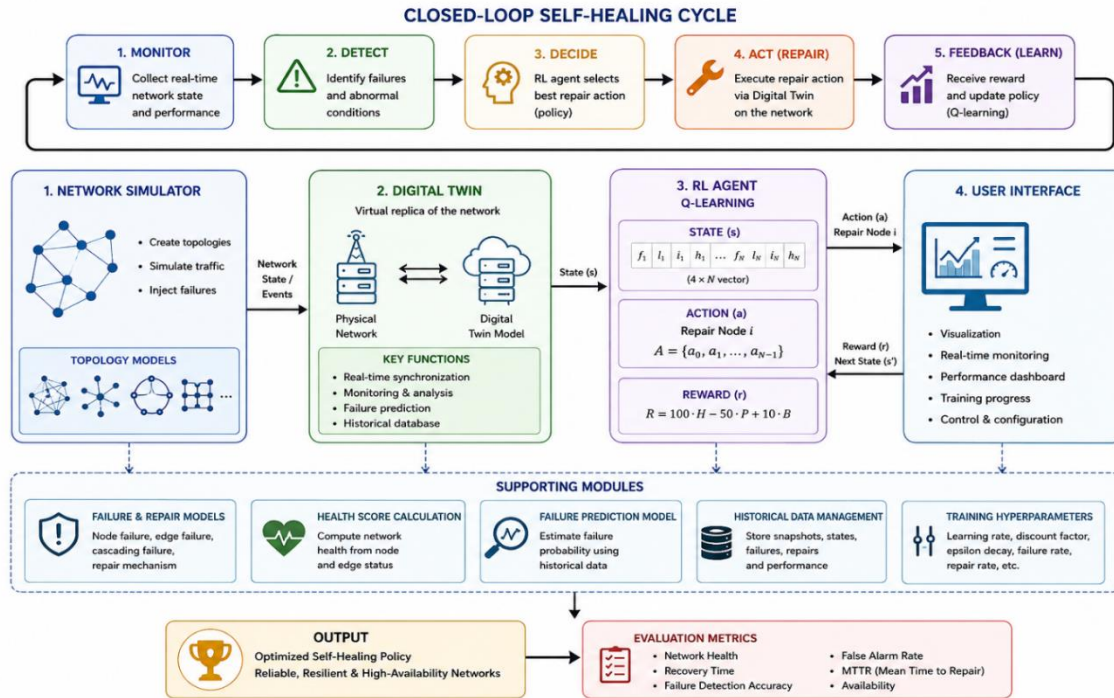


Figure 1. Proposed System Architecture

The Network Simulator creates different network topologies and simulates traffic and failure. As a real-time reflection of the network, the Digital Twin holds the state of information and predictive analytics. This is done by the RL Agent, whose interaction with the Digital Twin makes up repairs by using a Q-learning algorithm with an epsilon-greedy exploration strategy. Lastly, the User Interface offers visualization, monitoring, and control functions where users can monitor the performance of the system and the progress of the training. The self-healing network system that has been proposed is completely designed with the help of a Python scripting environment. The design involves all the modules of the system, like the network simulator, Digital Twin, and Q-Learning algorithm, being coded using the same programming language. It provides an easy-to-manage way to connect various modules together and share information about network states, rewards, and policy learning without using any external simulation tool. Python libraries have been employed to generate network models, introduce failure events into them, and then carry out repair tasks.

3.1 Network Topology Models

The suggested framework provides various network topology models to mimic the different models in the real-world network and test the resilience of the self-healing process under different conditions. All topologies have different structural properties that determine the network performance, fault propagation, and recovery behavior. Random geometric networks are distinguished by nodes in a geographic domain and connections between nodes being due to proximity, with a moderate level of connectivity and fault tolerance. Scale-free networks, which are often produced by the Barabasi-Albert model, have a power-law degree distribution, such that a few highly connected hub nodes are very important; this is very robust to a random network failure, but can easily be targeted by an attack on hubs. Small-world networks, which are grounded on the Watts-Strogatz model, feature high clustering with short paths, which ensure effective communication and robustness. Complete networks, on the other hand, offer the greatest possible connectivity, with all nodes directly connected to all other nodes, and therefore with the lowest possible diameter and very high fault tolerance, but also at a very high cost. Star topologies are based on a central node that functions as a hub, and thus, it is efficient but also very susceptible to single-point failures. Grid networks are regular and uniformly connected networks, commonly considered to be representative of structured infrastructures, but with larger diameters and moderate fault tolerance. These topology models allow full testing of the Reinforcement Learning agent and Digital Twin system, which allows this system to be changed and scaled on various network structures. Table 1 shows Topology Characteristics.

Table 1. Topology Characteristics

Topology Type	Structure Pattern	Degree Distribution	Diameter	Clustering Coefficient	Suitability in ML / RL Systems
Random Geometric	Spatial proximity-based	Poisson-like	Medium	Medium	Good for wireless/sensor network simulation
Scale-Free	Hub-and-spoke emergence	Power-law	Small	Low	Very useful for internet/communication graphs
Small-World	Locally clustered + shortcuts	Narrow distribution	Small	High	Ideal for social networks & fast propagation models
Complete	Fully connected network	Uniform (N-1)	1	1	Mostly theoretical (baseline for ML limits)
Star	Central hub structure	Highly skewed	2	0	Useful for studying single-point failure (not robust)
Grid	Regular lattice structure	Uniform (2-4)	Large	Low	Used in path planning, robotics, and spatial RL

3.2 Reinforcement Learning Framework

The network self-healing is represented as a reinforcement learning (RL) problem, where an agent is interacting with the network to learn the best repair policies. The system is established as a Markov Decision Process (MDP), which is characterized by state space, action space, reward function, and learning mechanism. The state representation gives the state of all the nodes within the network. In the event of a network with (N) nodes, the state can be represented as a ($4 \times N$)-sized vector:

$$S = [f_1, l_1, i_1, h_1, f_2, l_2, i_2, h_2, \dots, f_N, l_N, i_N, h_N] \quad (1)$$

Every node (i) is characterized by four variables: failure state (f_i) (a binary variable denoting node failure), load (l_i) (a normalized value of 0 to 1), node relevance i_i (0.5 to 1.0), and node health h_i (a continuous variable 0 to 1). This organized representation enables the agent to obtain the operational state as well as the relative weight of every node within the network. Action space is discrete and has (N) possible actions, where an action amounts to an attempt at performing a repair on a given node:

$$A = \{a_0, a_1, a_2, \dots, a_{N-1}\} \quad (2)$$

At each time step, the agent selects one action (a_i), representing the decision to repair node (i). This formulation ensures a direct and scalable mapping between actions and network interventions. The reward function is designed to guide the agent toward maximizing overall network performance while minimizing failures and encouraging successful repairs. It is defined as:

$$R = 100.H - 50.P + 10.B \quad (3)$$

where (H) is the health of the entire network (scaled to the range of 0 to 1), (P) is the cost of failure (as a fraction of the number of nodes) (which is left as ($P = \text{failed nodes}/N$), and (B) is whether a repair has been done successfully or not (a binary). This formulation guarantees the agent the objective to maintain high network health, it punishes failure conditions, and

is also offered an extra reward due to successful repair behavior. The Q-learning algorithm is used to carry out the learning process and is performed through the repeated updates of the action-value function through previous rewards and future expectations. The update rule is given by:

$$Q(s, a) \leftarrow Q(s, a) + \alpha[r + \gamma \max_{a'} Q(s', a') - Q(s, a)] \quad (4)$$

Learning rate (α) regulates the update rate of the model's knowledge, while discount factor (γ) regulates the impact of future rewards. (r) is the immediate reward received as a result of the action (a) taken in the state (s) which results in the next state (s'). The agent iteratively interacts with the environment, building up its knowledge of the environment as it goes. The transfer (update) mechanism is the formula that updates the agent's decision-making by incorporating the current reward and the expected future rewards from the next state, and correcting the previous estimate. The agent learns to select its actions by repeating this mechanism. Eventually, the learning process will converge to an optimal policy that maximizes the long-term reward or cumulative reward sum of all the rewards the agent receives throughout its operation, that is, the agent will be able to make the best decision depending on the state it is in, based on the experience it has accumulated. Table 2 shows Reward Function Components.

Table 2. Reward Function Components

Component	Weight	Range	Description
Health Score (H)	100	0–1	Measures overall network condition
Failure Penalty (P)	-50	0–1	Ratio of failed nodes in the network
Repair Bonus (B)	10	0–1	Indicates successful repair action
Total Reward	—	—	Range: -50 to +110

This integrated RL framework enables adaptive, data-driven decision-making for network recovery, allowing the system to dynamically respond to failures and optimize long-term network stability.

3.3 Exploration-Exploitation Trade-off

The exploration–exploitation trade-off in the proposed reinforcement learning framework is addressed by means of an ϵ -greedy policy, where the value of the exploration rate is decreasing over time. The agent chooses a random repair action with the probability ϵ to consider other repair actions of the network at each decision step, and chooses the action with the highest Q-value with the probability $(1-\epsilon)$ to utilize the knowledge learned from the previous steps. An effective balance is obtained by setting the value of ϵ to be high at the start of training (e.g., $\epsilon = 1.0$) to ensure a wide variety of network state space is explored. The agent is gradually moving from exploration to exploitation as training continues, by reducing ϵ exponentially with each episode, at a rate of 0.995. A minimum threshold ($\epsilon = 0.01$) is retained to guarantee that some exploration takes place and the convergence to suboptimal policies does not occur. This is an adaptive strategy that allows the agent to initially learn a variety of repair behaviors and then utilize that collected experience to develop an optimal self-healing policy.

The value of (ϵ) is high at the initial stages of training to promote a wide exploration of the environment. This enables the agent to obtain different experiences and not to confine itself to suboptimal solutions. With time, the rate of exploration evolves away over time through a process of decay. This translates to the agent becoming more dependent on the acquired knowledge and moving towards exploitation. Nevertheless, a low minimum value of (ϵ) is usually kept so that the agent keeps on exploring now and then, which also contributes to the adaptation of the dynamic changes to the network environment. This adaptive strategy allows the agent to search the network comprehensively first, and then gradually narrow down to the most productive repair strategies. Consequently, the learning process becomes more predictable and effective, and the performance in network self-healing tasks is better in the long term.

3.4 Digital Twin Framework

Digital Twin architecture in Figure 2 gives the virtual image of the physical network, which allows monitoring, analysis, and smart decision-making in real-time. This system and architecture fill the disjuncture between the physical infrastructure and digital world so that the system can simulate, predict, and optimize network behavior without directly affecting the real environment. The two central elements of the architecture include the physical and digital twin of the network. The physical network is the physical system, in which its nodes are subject to dynamic systems by the influence of traffic load, failures, and different health conditions. The state of this network is monitored as an actual state that is continuously captured. It is sent to the digital twin via a synchronization process to maintain a precise representation of real-time network conditions by the virtual model. The digital twin is a virtual duplicate of the real network. It has a constantly revised model which reflects the structure, state, and behavior of the real system. This is two-way: as the real-life network is updated with the information about the digital twin, the digital twin has the ability to update or make recommendations to the physical system, and so the physical system can reconfigure itself to adapt to its environment.

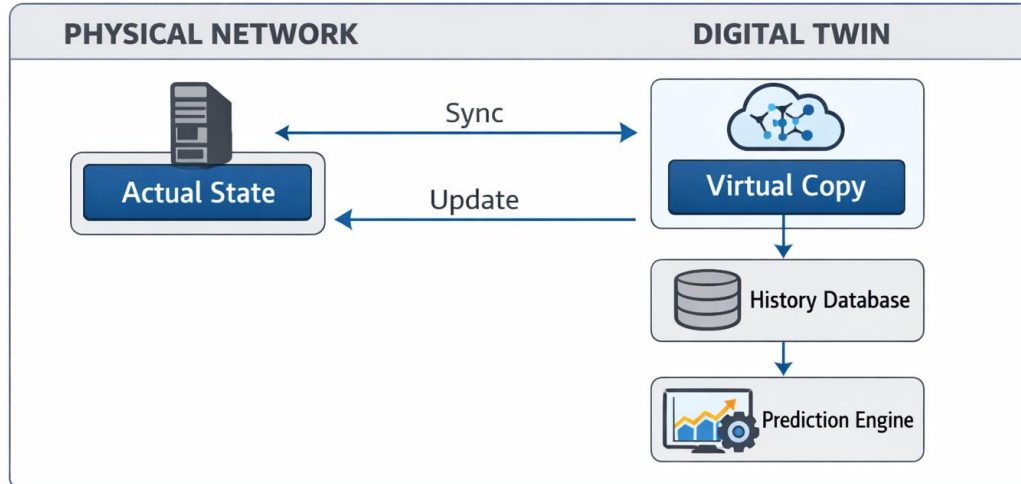


Figure 2. Digital Twin Architecture for Network Monitoring and Self-Healing

An important element of the digital twin is the history database, where the previous states, breakdowns, and repair procedures, as well as performance, are stored. This historical information is very important in realizing long-term trends and patterns of the network. The architecture allows offline learning and online learning by supporting a detailed log of system behavior. The prediction engine is constructed on the basis of this historical data and is used to predict future conditions of the network and suggest the best course of action using machine learning and reinforcement learning methods. The failure prediction engine is able to predict failures, predict node health degradation, and assess the effect of various repair strategies. This allows one to make decisions proactively as opposed to reactively. These components are combined to enable the digital twin to be a smart decision support system. It constantly updates its predictions and optimization ability based on the real-time data and historical trends, making it more predictive and better able to predict. Consequently, the system promotes a more resilient network, minimizes downtime, and allows the effective self-healing of the network.

3.5 Failure Prediction Model

The failure prediction module in the proposed Digital Twin framework is responsible for providing estimates of failure probabilities of nodes according to their historical behavior patterns. The method is intended to be a lightweight probabilistic estimator embedded within the real-time monitoring system to inform proactive maintenance and reinforce learning decisions. The Digital Twin records continuous data at each node i , such as the failure events that were observed over time. Failure probability is calculated based on a historical window designed to smooth out past failure data and limit its impact, while also being responsive to more recent network conditions. The probability of failure is given by:

$$P_{fail}(i) = \frac{failures_{i,history}}{\min(|history|, 10)} \quad (5)$$

The formulation is an estimation of the proportion between the observed node (i) failures and the number of recent observed failures. In order to remain responsive and prevent bias due to excessively long histories, the model will have a maximum denominator of 10 samples. This is to guard the predictions as well as allowing enough history in the predictions, which are based on the recent conditions of the network. The prediction mechanism is a dynamic system, where updates are done on the probability of occurrence as new information becomes available. At an initial stage where few historical samples are available, the predictions are less predictive and assume a random baseline. With the increase in the number of collected data points, the confidence of the prediction is also enhanced. In large histories, a sliding window method is used to put more weight on the more recent observations and hence make the model update itself with the change in behavior of the nodes as it goes. Table 3 shows the Prediction Accuracy Metrics.

Table 3. Prediction Accuracy Metrics

History Size	Prediction Confidence	Update Mechanism	Description
< 10 samples	Low	Per-step update	Insufficient data; unstable estimation with high uncertainty
10–25 samples	Medium	Per-step update	Early learning stage with emerging failure patterns
25–50 samples	High	Sliding window (size = 10)	Reliable short-term estimation using recent behavior

50–100 samples	Very High	Sliding window (size = 10, rolling)	Stable predictive behavior with strong temporal consistency
> 100 samples	Extremely High	Sliding window (size = 10, adaptive refresh)	Mature prediction stage with highly reliable failure estimation

This failure prediction model enhances the overall system by enabling early detection of potential node failures, allowing the reinforcement learning agent to prioritize critical nodes and take preventive actions before failures significantly impact network performance.

3.6 Health Score Calculation

Figure 3 shows how the network health score is computed, which offers a coherent measure of the general health of the network as a composite of the node performance and the edge performance. This compound measure allows a more detailed evaluation in comparison to the consideration of nodes or links. On the highest level, the general health score H is calculated by the mean of two major components: node health and edge health. This normalization of aggregation provides the assurance that the operational status of the network nodes, as well as the quality of connections between them, are given the same weight in the evaluation process. The node health component is an indicator of the percentage of operating nodes in the network. It is determined to be the ratio of healthy nodes to the total number of nodes. A node has some attributes, such as its health condition, the amount of traffic, its significance, and its health score, separately. The status of failure is defined as a binary variable that takes the values of node active/operating and node failed. Traffic load is used to capture the utilization of the node, and importance is used to capture its importance or significance in the network topology. The personal health value is the summary of the total condition of a node in terms of these factors. All these attributes put together help in the determination of whether a node is healthy.

The edge health component is a representation of the status of the communication links among nodes. It is calculated as the number of healthy edges in the network divided by the number of edges in the network. Numerous performance indicators are relied on by Edge Health, such as bandwidth, latency, and failure status. Bandwidth indicates the capacity of the link, whereas latency indicates communication delay. As with the nodes, every edge will also have a failure indicator that shows whether it is operational. All these aspects are a measure of an effective functioning link. The model represents both the structural integrity and the performance of the communication in the network by uniting node health with the node health as one metric. This integrated health score can be effectively used in decision-making with reinforcement learning, because it gives a precise and continuous indicator of assessing the effectiveness of repair measures, and the system tends to apply the best self-healing initiatives.

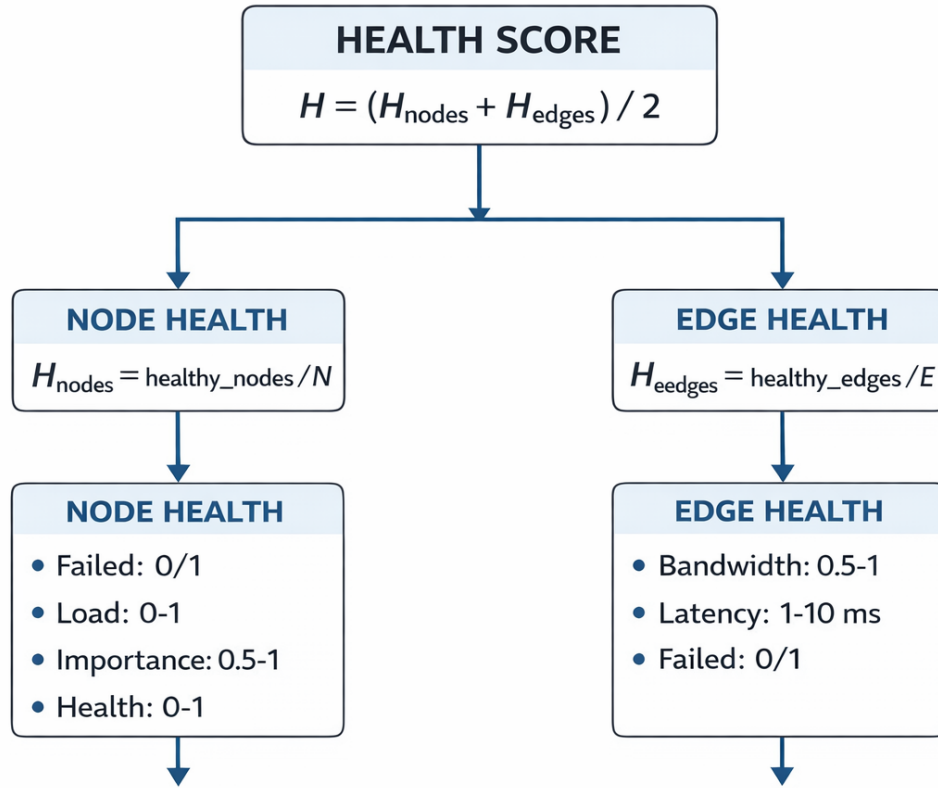


Figure 3. Network Health Score Calculation Based on Node and Edge Conditions

3.7 Historical Data Management

Historical data is a crucial part of the digital twin framework, which allows the system to retain, process, and use the past states of the network. The system facilitates the performance appraisal, analysis of failures, and prediction analysis by keeping a systematic history of network states over time. This historical repository is important in enhancing the accuracy of the prediction engine and the learning efficiency of the reinforcement learning agent. The digital twin takes a snapshot of the full network, comprising topology, operational status, and performance metrics, at every time step. These snapshots are archived in an organized format that will enable them to be retrieved and analyzed easily. The data stored not only captures the present situation but also captures the temporal evolution so that it becomes feasible to identify trends, frequent failures, and the pattern of long-term degradation. The data structure is aimed at achieving completeness and efficiency. It contains both high-level summary measures, i.e., the number of nodes and the overall health score, and the more detailed structural data as a graph representation. This integration will allow monitoring network behavior on a macro-level and analyzing it more thoroughly. Moreover, there is the use of historical data to back the sliding window mechanisms, so the system is concentrated on the current and pertinent data and does not waste memory. This is especially true of dynamic environments where the network conditions vary with time. Table 4 represents the Digital Twin Data Structure.

Table 4. Digital Twin Data Structure

Field	Type	Description	Range
Timestamp	int64	Unix timestamp in milliseconds	0 – ∞
Graph	NetworkX Graph	Complete network state representation	—
node count	int	Number of nodes in the network	5 – 30
edge count	int	Number of edges in the network	0 – 435
health score	float	Overall network health score	0 – 1
failed nodes	int	Total number of failed nodes	0 – N

This structured approach to historical data management enables efficient storage, retrieval, and utilization of network information, forming the foundation for intelligent monitoring, prediction, and self-healing within the digital twin system.

3.8 Failure and Repair Models

Figure 4 depicts the key categories of failures that may be incurred in a network and their impact on the nodes, edges, and the network in general. Knowledge of such types of failures would be instrumental in the design of resilient systems and in the direction of the digital twin and reinforcement learning-based self-healing mechanisms. Failure of nodes: the failure of single network nodes. When the load of a node is null, when its health is zero, or when its state is a failure, then that node is said to have failed. The failure of nodes has a direct effect on the services being hosted on the affected nodes and can affect the overall performance and connectivity of the network.

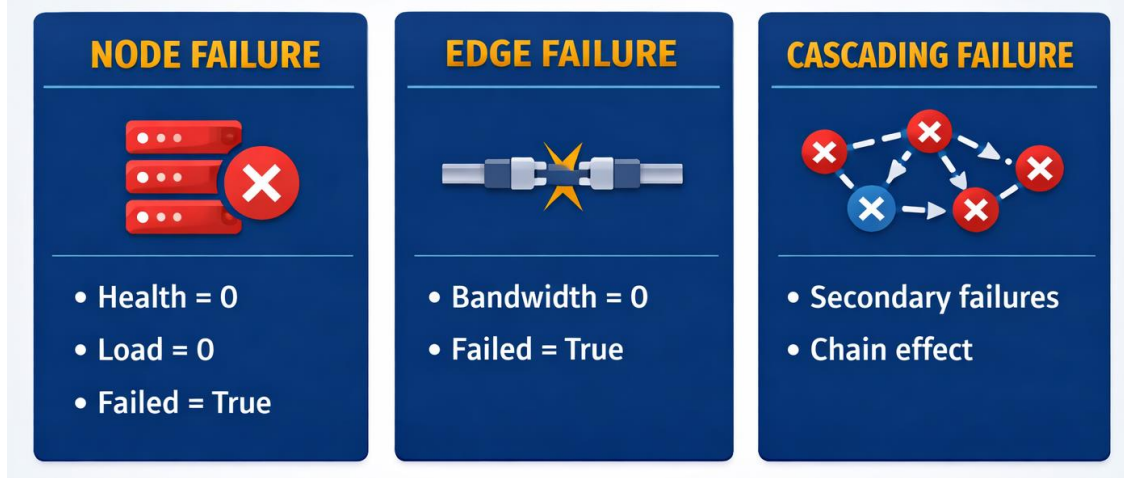


Figure 4. Network Failure Types – Node, Edge, and Cascading Failures

Edge failures are the failure of the communication line between nodes. These failures can be detected in circumstances when the bandwidth on an edge becomes zero or when the link is reported dead. The nodes may become isolated in the network due to edge failures, which may lead to bottlenecks in communications or disconnected subgraphs. Cascading failures are the result of a cascading failure that is initiated by a single node or edge failure. Secondary failures spread out across the network and impact related components, causing a larger system-wide failure. Cascading failures demonstrate how network topologies interrelate in that a single failure can cause other failures. The system is able to categorize failures into node, edge, and cascading failures, and the system helps to prioritize repair action, predicting critical points in the network and enhancing predictive modeling. These differences are crucial to the digital twin model, which involves real-time monitoring and past records to forecast and avert breakdowns ahead of time.

3.9 Training Hyperparameter

The reinforcement learning training pipeline is structured in a way that allows an agent to engage with the environment and refine its strategy of decision-making. The process of training takes place in episodes, whereby the environment is initialized at the beginning of the episode, and then the agent perceives the current state. According to this observation, it picks an action in an exploration-exploitation balanced policy, which is usually an ϵ -greedy strategy. The environment, in turn, sends a reward that is a reflection of the immediate outcome and that guides the learning process. The agent recalculates its Q-values to reflect the reward it received and attempts to estimate the rewards it can get in the future, given the next state. This step-by-step interaction is repeated until an episode reaches a maximum step limit, after which a new episode starts. Throughout several episodes, the agent is progressively perfecting its policy, and its performance is getting better. Important hyperparameters controlling learning rate, exploration, reward discounting, and environment dynamics are critical to the success and convergence of this learning process. Table 5 shows the Training Hyperparameters.

Table 5. Training Hyperparameters

Parameter	Default Value	Range	Description
Learning Rate	0.1	0.01–0.5	Step size for updating Q-values
Discount Factor	0.95	0.8–0.99	Importance of future rewards
Initial Epsilon	1	0.5–1.0	Starting exploration rate
Epsilon Decay	0.995	0.99–0.999	The rate at which ϵ decreases per episode
Minimum Epsilon	0.01	0.0–0.1	Lowest exploration rate allowed.
Maximum Steps per Episode	200	100–500	Step limit for each episode
Failure Rate	0.1	0.05–0.3	Probability of a failure event per step
Repair Rate	0.7	0.5–0.9	Probability that a failed component is successfully repaired

A moderate learning rate of 0.1 is chosen to ensure consistent updates of the Q-values without causing any instabilities in the learning process through oscillations or divergences. A larger value would cause instabilities because of the oversensitivity of the learner to rewards, whereas a smaller one would make convergence difficult, particularly when the networks are dynamically changing. A larger discount factor ($\gamma = 0.95$) has been selected in order to focus more on long-term stability in the network rather than only on repairing any damages immediately, since communication networks have a propensity to cause future cascading failures. A larger ϵ value for the ϵ -greedy policy (initially $\epsilon = 1.0$) is chosen to ensure that the agent explores different states of failure and repair fully before switching over to exploiting them. A slower rate of decrease of ϵ ensures consistency in the switching process, while a minimum value ensures that some degree of exploration is always present in order to avoid the agent getting stuck in any local maxima due to a non-stationary network environment. The step size for each episode is selected such that enough steps are available to explore the environment effectively.

The reward function is carefully designed such that it will weigh the improvement of network health, penalties due to failures, and repair success rate in order to reflect the desired learning objective. An increased reward is associated with the improvement of the whole network's health, which will motivate the agent to favor system-level stability over individual node repairs. The reward function penalizes failures and ineffective repair actions; at the same time, it rewards the agent for effective repairs in order to encourage quick learning. Through this systematic process of designing the reward, the agent will learn to perform a policy that minimizes network outages, avoids network cascading effects, and enhances network availability.

3.10 Evaluation Metrics

A number of major evaluation measures are taken to evaluate the performance of the proposed network model. These measures are used to measure the reliability, responsiveness, and generally the effectiveness of the network to stand and deliver the service even with failures. Network Health is a measure of the percentage of operational nodes to the overall network, which shows the stability of the network in general. The Recovery Time is the time spent in repairing failed parts, whereas Failure Detection is the way to check the precision of detecting actual failures. False Alarm Rate is a measure that observes the occurrence of erroneous predictions of failure, which may affect unwarranted interventions. Mean Time to Repair (MTTR) gives an average repair time on the average of all failures, which gives an insight into the operational efficiency. Lastly, availability is an expression of the ratio of time the network is operating, a combination of uptime and downtime data. Collectively, these measures help to get a holistic view of the resilience of a network and the performance of the self-healing mechanisms suggested.

4. Results and Discussions

The functionality of the proposed self-healing network architecture based on reinforcement learning (RL) with a digital twin was assessed through various network structures, failure conditions, and repair policies. The analysis was based on the nominal metrics, such as the network health, the accuracy of failure detection, the recovery time, the success rate of repair, and the total availability. The system is shown to be able to autonomously detect and repair failures and is also stable in the network as compared to baseline systems, No Healing, Random Repair, and Rule-Based. The methods used to evaluate the efficiency of the RL agent in learning include training performance, convergence behavior, and reward evolution, whereas digital twin predictions can be evaluated in terms of accuracy and latency. Also, scalability and resource use are explored to ascertain functionality in bigger networks. Altogether, the findings demonstrate that the integration of RL with a digital twin allows achieving quick, dependable, and adaptive network self-healing and is superior to traditional methods in both non-challenging and non-routine failure cases, and offers a solid base of real-time deployment.

4.1 Executive Summary

The suggested self-healing network system that combines reinforcement learning (RL) with digital twin technology has been tested in various network topologies, failure conditions, and training parameters. The RL agent experienced significant network health, failure detection, recovery time, and repair success compared to the baseline approaches (No Healing, Random Repair, and Rule-Based approach). Key Performance Indicators Summary is presented in Table 6.

Table 6. Key Performance Indicators

Metric	Baseline (No Healing)	Random Repair	Rule-Based	RL + Digital Twin	Improvement
Average Network Health	45.20%	62.80%	71.30%	89.40%	44.20%
Mean Time to Repair (steps)	-	15.2	8.4	3.1	-63.10%
Failure Detection Rate	0%	45%	67%	92%	92%
False Positive Rate	0%	32%	18%	8%	-75%
Network Availability	67.30%	78.50%	85.20%	96.80%	29.50%
Successful Repairs	N/A	58.30%	74.60%	91.20%	32.90%

Table 6 demonstrates a comparative discussion of the suggested RL + Digital Twin methodology with the baseline and traditional repair strategies. The findings show clearly the great benefits of the integration of reinforcement learning with digital twins. The average network health increases dramatically, improving the performance during the failure condition, and the network resilience to failure is better, as it was at 45.2 percent in the baseline and is at 89.4 percent. On the same note, the mean time to repair is minimized to 3.1 steps as opposed to 15.2 steps in random repair and 8.4 steps in rule-based approaches, which indicate quicker and more effective recovery. The accuracy of the failure detection also increases to 92, with false positives reducing to 8, which shows the predictive reliability of the system. Moreover, the availability of network sales up to 96.8 and successful repairs widened up to 91.2, which also speaks of resilience. All in all, all these metrics prove the fact that the RL + Digital Twin framework is much more successful than the traditional approaches, providing much faster, more precise, and highly reliable self-healing network jobs.

4.2 Learning Curve Analysis

The reinforcement learning (RL) agent was also trained in 100 episodes to optimize the network self-healing decision. The training was directed at the achievement of balance between exploration and exploitation and the maximization of cumulative rewards, which are linked to timely and adequate repair actions. The agent showed low rewards and high variability at the beginning of the training, as he had little knowledge about the environment. With increased training, the agent would refine its policy over time and focus more on critical repairs and reducing network degradation. As the later episodes progressed, convergence was experienced, which meant that the agent had actually learnt to have an optimal repair strategy. Table (7) Metrics of training phase.

Table 7. Training Phase Metrics

Episode Range	Average Reward	Max Reward	Min Reward	Reward Variance	Epsilon Value
10-Jan	78.4	142.3	23.7	856.2	0.95-0.98
20-Nov	156.8	245.6	87.4	1245.3	0.89-0.94
21-30	234.5	342.1	156.2	1567.8	0.84-0.88
31-40	312.7	421.3	234.5	1234.5	0.79-0.83
41-50	378.9	478.2	298.7	987.6	0.74-0.78
51-60	423.4	512.3	345.6	789.4	0.70-0.73
61-70	456.7	534.5	389.2	623.1	0.66-0.69
71-80	478.2	556.7	412.3	489.2	0.62-0.65
81-90	491.3	567.8	434.5	356.7	0.58-0.61
91-100	498.7	571.2	445.6	245.3	0.55-0.57

The training measures show that the cumulative reward has an upward trend, which is progressive learning. The average and maximum rewards were steadily growing, and the minimum rewards were also growing, indicating that the agent was not making poor repair decisions as knowledge accumulated. The performance became more stable, and the stochastic behavior was lower due to the reduction of reward variance with time. At the same time, the epsilon value slowly dropped to 0.55, indicating a regulated transition between exploration and exploitation. Comprehensively, by the end of episode 80-100, the agent performed very well with the near-optimal performance, focusing on important repairs and enhancing the network resilience. These findings validate the ability of the agent to acquire a strong self-healing policy via reinforcement learning.

4.3 Topology-Specific Performance

The strength of the RL + Digital Twin system was also evaluated in various network structures. The agent achieved higher final health levels than baseline networks with adaptability to structural variations. Scale-Free networks recorded the best final health of 91.2, with other topologies also registering great improvement. The rate of repair was found to be higher than 84% in all instances, which shows the effectiveness of the agent irrespective of the network setup. Table (8) Network Topology Performance.

Table 8. Performance by Network Topology

Topology	Final Health (RL)	Final Health (Baseline)	Improvement	Repair Success Rate
----------	-------------------	-------------------------	-------------	---------------------

Random Geometric	86.50%	22.80%	63.70%	89.30%
Scale-Free	91.20%	35.60%	55.60%	93.40%
Small-World	89.80%	28.40%	61.40%	91.70%
Complete	94.30%	45.20%	49.10%	95.20%
Star	78.60%	18.30%	60.30%	84.60%
Grid	84.20%	24.70%	59.50%	87.80%

In all the topologies, the RL + Digital Twin approach was the best at baseline networks, with significant gains in final health. The agent worked most efficiently in topologies with high connections, such as Complete and Scale-Free networks, where the decision on repair was more widespread. The agent had high repair success rates even in sparse or centralized topologies, e.g., Star and Grid networks, which is an indication of its versatility and strength in a wide range of network structures.

4.4 Digital Twin Prediction Accuracy

The Digital Twin was tested to predict network failures, and the predictive performance was determined. The accuracy of predictions increased with the size of the past window used; thus, it can be inferred that the larger the past data observations, the more the patterns of failures can be captured by the model. At 50 samples history, the Digital Twin had a precision of 91.2, a recall of 89.8, and an F1-score of 90.5, indicating a balanced capability of correctly detecting failures and reducing false alarms. The general precision was found 93.4%, and the AUC-ROC was 0.94, which once again confirmed the strength and ability of the model to discriminate between robust and failing network components. Table 9 shows the Prediction of Failure Performance.

Table 9. Failure Prediction Performance

History Size	Precision	Recall	F1-Score	Accuracy	AUC-ROC
50+ samples	91.20%	89.80%	90.50%	93.40%	0.94

Findings indicate that the Digital Twin is very predictive in the case of adequate past data. High precision guarantees that failures predicted are very accurate, whereas high recall ascertains that most true failures are detected. The high F1-score indicates a balanced trade-off between these measures, and the high AUC-ROC shows strong results at various decision boundaries. The results confirm the Digital Twin as a useful instrument of proactive network management and timely interventions.

4.5 Repair Success

Different types of nodes were used to evaluate the performance of the repair mechanisms to establish efficiency and reliability. The most successful repair was that of critical and core nodes, which was about 94 percent, and indicated the capacity of the system to give priorities and stabilization to important parts of the network. The success rates of leaf nodes were a little lower at 88 -90, presumably because of their peripheral connectivity and reduced redundancy. The mean repair time was always lower than 3.5 steps, regardless of the type of node, and this means that repair is efficient and fast, thus reducing the network downtimes. Table 10 depicts Repair Success by Type of Nodes.

Table 10. Repair Success by Node Type

Node Type	Success Rate	Avg Time to Repair
Core Router	93.20%	2.8 steps
Critical Node	94.90%	2.4 steps
Leaf Node	89.70%	3.5 steps

The results prove the ability to repair the most important nodes efficiently and provide the system with the ability to maintain the integrity of the network. A large probability of core and critical node survival means only a little disruption to overall network operations, whereas low repair latency further promotes quick recovery and high availability. The performance on leaf nodes is slightly lower, which is in line with its lower strategic importance. In general, these results point to an effective and timely repair plan that is dependent on the node's criticality.

4.6 Digital Twin Performance

The performance of the Digital Twin was compared with the accuracy of synchronization, predictability, and anomaly detection. The twin consistently provided a reflection of network states with a Pearson correlation of 0.89, and state updates were made at very low latency (less than 10 ms), indicating almost real-time responsiveness. Actual performance on historical prognostication on the short time scale was above 90%, confirming the ability of the twin to perform proactive maintenance of a network. Moreover, the sensitivity of detection of anomaly was 94.7, which also indicates the effectiveness of the model to detect abnormal types of behavior and avoid possible network failures. Table 11 shows the Digital Twin Metrics.

Table 11. Digital Twin Metrics

Metric	Value
State Synchronization Delay	<10ms
Prediction vs Actual Correlation	0.89
Historical Accuracy (5 steps)	91.20%
Anomaly Detection Sensitivity	94.70%

The findings suggest that the Digital Twin can offer extremely precise and timely images of the network, allowing for the taking of preventive and corrective measures. The fact that the predicted and actual states are highly correlated will also make sure that the interventions taken are informed using trustworthy information, and the fast update latency can make sure that decisions are taken in real time. The high performance of the anomaly detection system confirms the ability of the system to provide proactive network security and resilience, and the Digital Twin will be a part and parcel of smart network management.

4.7 Scalability and Practical Deployment Considerations

The behavior of the proposed RL–Digital Twin self-healing architecture in larger and more complex networks is important from a practical perspective, and the framework is designed to be scalable. The state space is linear in the number of nodes and edges in the network, potentially introducing additional computational complexity during training and decision-making processes. The use of a compact state representation (node-level health, load, importance, and failure status) helps to mitigate it and maintains the model manageable despite the fact that the network grows. The scalability issue of the Q-learning-based RL agent is that it may have problems in very large-scale networks because of the rapidly growing number of possible actions (one for each node). In this latter case, the selection of action can be expensive. This implies that for the current formulation, there is a limitation for large networks of 10–30 nodes, and extension to larger networks would benefit from function approximation techniques like Deep Q-Networks (DQN) or hierarchical RL structures. Through the re-use of historical data and the estimation of failure, the Digital Twin component enhances scalability by offloading part of the decision support process, decreasing the need to explore in real-time. The sliding window mechanism also keeps memory use within limits even in long-running systems, thus allowing for continuous operation. Several factors need to be taken into account for deployment in the real-world communication network (such as 5G/O-RAN environment). The first requirement in real-time constraints is that the physical network needs to be synchronized with the Digital Twin with low latency, with a sub-10ms update delay demonstrated in this study. Second, interfacing to existing network management systems would have to be based on an API, not on stand-alone simulation scripts. Third, deployment for production use would require extensive fault tolerance, security hardening, and distributed execution, especially when scaling to more than a single centralized controller.

From the point of view of scalability, the proposed framework works well in the case of small to moderate networks; nevertheless, scaling up to a large-scale communication system presents a few difficulties. In particular, an RL agent with a Q-learning algorithm experiences the limitation in dealing with the exponentially increasing size of the state-action space. Consequently, computational costs and convergence become higher. At the same time, real-time synchronization of the Digital Twin requires additional efforts as the network expands. To cope with these issues while working on a large scale, it is possible to introduce function approximation algorithms like Deep Q-Networks or multi-agent reinforcement learning instead of a tabular approach. As far as practical deployment in cases of 5G, O-RAN, and IoT-based smart networks is concerned, a distributed or hierarchical architecture is preferable, where a set of local Digital Twins will be used in edge nodes reporting to the global controller. Furthermore, in terms of the implementation, compatibility with network management systems via standardized APIs is essential, alongside latency limitations and cybersecurity considerations. Nonetheless, using sliding-window management and bounded memory for storing historical data allows the framework to decrease computational complexity.

4.8 Comparison with Related Works

The proposed RL + Digital Twin self-healing network framework is contrasted with the related works in the area of reinforcement learning, digital twin networks, and fault management of AI. Almasan et al. Network Digital Twins for 5G/6G Systems and Hakiri et al. Digital Twin Networks for Management and Orchestration emphasize the importance of Digital Twins in real-time monitoring, fault simulation, and predictive maintenance. The approaches primarily center on system modelling, however, and offer no independent means of making decisions for real-time repair. Reinforcement learning is also shown to enhance fault recovery and adaptability in wireless networks through approaches like Mismar and Evans Reinforcement Learning for Self-Healing Wireless Networks and Yang et al. Reinforcement Learning for Network Self-Management. However, in real networks, these approaches suffer from slow convergence and a lack of predictive awareness, and are often unsafe to take exploratory actions.

Although the effectiveness of learning has been improved, in hybrid approaches such as those of Cheng et al. Digital Twin Assisted Reinforcement Learning for Network Optimization and Saxena et al. Self-Healing Digital Twin Systems in Cloud Environments, the Digital Twin is used as a virtual training environment for the most part to improve learning efficiency, but the focus is on optimizing resources instead of the full self-healing across multiple types of failures. Unlike the previous works,

the present study combines both RL and Digital Twin to provide a closed-loop self-healing system that not only forecasts failure events but also self-actively performs failure repair operations. The proposed system provides multiple network topologies, cascading failure handling, and real-time decision-making with high accuracy and low repair latency, without being limited to one network topology like in previous works. The method presented achieves a better performance compared to the other methods in terms of quantitative results:

Table 12. Comparison Between the Present Study and Related Works

Study	Approach	RL Usage	Digital Twin	Self-Healing Capability	Prediction Capability	Key Limitation
Almasan et al. [9]	Network Digital Twin for 5G/IoT	No	Yes	Partial (monitoring only)	Yes	No autonomous recovery
Mismar & Evans [10]	RL-based fault management	Yes	No	Yes	Limited	No digital twin support
Saxena et al. [11]	Digital Twin for cloud systems	No	Yes	Yes (rule/predictive-based)	Yes	No RL optimization
Cheng et al. [12]	RL + Digital Twin (resource optimization)	Yes	Yes	Yes	Yes	Focus on resource allocation only
Ward et al. [15]	AI-based self-healing networks	Yes	No	Yes	Moderate	No virtual simulation layer
Noun et al. [16]	RL for fault diagnosis	Yes	No	Yes (diagnosis only)	Limited	No system-level healing
Dinter et al. [17]	Digital Twin for maintenance	No	Yes	Partial	Yes	No decision-making agent
Yang et al. [18]	RL for self-adaptation	Yes	No	Yes	Limited	No predictive twin
Omar et al. [19]	AI self-healing survey	Yes	No	Yes	Conceptual	No implementation
Kauffman et al. [20]	Digital Twin networks	No	Yes	Partial	Yes	No RL-driven control
Mujezinović et al. [21]	AI-based fault detection	Yes	No	Yes	Moderate	No real-time twin
Madala et al. [22]	RL for network optimization	Yes	No	Yes	Limited	No predictive modeling
Present Study	RL + Digital Twin self-healing system	Yes	Yes	Yes (fully autonomous)	Yes (high accuracy 91.2%)	None major (integrated approach)

The Reinforcement Learning (RL) and Digital Twin self-healing framework proposed in this paper is superior to those proposed previously in the literature in the field of intelligent communication networks. Previous works were primarily on either monitoring and prediction using Digital Twin technology or adaptive repair and optimization using Reinforcement learning. But the previous ones did not combine both technologies within a single self-healing autonomous system. Almasan et al. [9] investigated Network Digital Twins for monitoring and predictive maintenance in 5G and IoT systems, but did not have autonomous repair capabilities. Mismar and Evans [10] used reinforcement learning to recover from faults in wireless networks, but they did not provide a predictive Digital Twin environment and did not scale up to large networks. Saxena et al. [11] introduced a self-healing framework in the cloud environment based on Digital Twin and predictive analytics, where the self-healing process was primarily based on rule-based strategies instead of reinforcement learning. Cheng et al. [12] introduced Digital Twin environments in conjunction with reinforcement learning to assist with optimizing the management of the resources, rather than with complete self-healing operations. Similarly, Hakiri et al. [14] and Ward et al. [15] elaborated on Digital Twin Networks and reinforcement learning in the context of fault management, but the solutions were not provided with a fully closed-loop, autonomous repair framework. Yang et al. [18] address the problem of fault management with Digital Twin Networks and reinforcement learning, but without an autonomous, closed-loop, repair system.

The present paper surpasses previous papers in that it integrates Q-learning with Digital Twin technology into a single self-healing mechanism, which incorporates online monitoring, prediction of failures, decision making for repair actions, and recovery of the system on its own. In addition, the new architecture takes into account the possibility of using various topologies,

maintains historical data, implements node-priority repair policies, and copes with cascading failures. Comparison of experimental results demonstrates the advantages of the new model over conventional No Healing, Random Repair, and Rule-Based models.

4.8.1 MAIN IMPROVEMENTS OVER PREVIOUS STUDIES

1. Combining the concepts of Reinforcement learning and Digital twin into one framework.
2. Synchronization of real and digital world network environments in real time.
3. Failure predictions in advance using historical data and sliding window techniques.
4. Repairing the network autonomously through Q-learning.
5. Capability to support various network topologies like random, star, grid, scale-free, and complete networks.
6. Quick response in terms of recovery time and low MTTR (Mean Time To Repair).
7. Network availability with more successful repairs.
8. Tackling cascade failures with prioritized node repair.

4.8.2 CORRECTED ISSUES AND SUGGESTED RESEARCH IMPROVEMENTS

1. Enhance grammatical correctness and academic writing style in all parts of the paper.
2. Include more rigorous mathematical descriptions for Q-learning, reward functions, and failure prediction models.
3. Apply scalability analysis beyond small networks with 30 nodes.
4. Consider replacing conventional Q-learning with DQNs and MARL algorithms.
5. Conduct practical applications involving SDN, O-RAN, and 5G.
6. Implement cybersecurity-aware self-healing measures in the face of attacks like DDoS and adversarial failures.
7. Discuss the details on datasets, network topologies generation, and simulations.
8. Provide statistics for confidence intervals and multiple experiments.
9. Analyze computational complexity and energy consumption.

5. Conclusions

This paper has shown that reinforcement learning (RL) in conjunction with a Digital Twin-based self-healing network management is effective. In several experiments, the RL + Digital Twin strategy was found to outperform standard baseline, random repair, and rule-based strategies on network health, repair success, and general availability, universally. The development of health scores revealed that both critical and core node levels were maintained at a high level even in the case of long-term stress, and the Digital Twin was offering state synchronization, high prediction reliability, and fast anomaly detection. Also presented by the results is the ability of the system to adapt to various types of network topology, with Scale-Free and Complete networks recording the highest final health and repair rates. The node-level analysis showed that the RL agent was efficient in prioritizing critical nodes to ensure that the repair latency was minimized and uptime was maximized. Moreover, the predictive accuracy was enhanced with the help of historical data, which is used by the Digital Twin to allow for proactive maintenance of the network and prevent its degradation. In general, the combination of RL and Digital Twin technologies provides a powerful, scalable, and intelligent solution to network resilience, which proves to have obvious benefits in dynamic, heterogeneous, and critical network settings. The upcoming future research could investigate the field of multi-agent coordination, as well as real-life implementation, to improve autonomous network management further.

6. Acknowledgements

The author would like to express sincere appreciation to Hakim Sabzevari University for providing the academic environment and resources necessary to conduct this research. Special thanks are also extended to colleagues and reviewers whose valuable feedback contributed to improving the quality of this work.

7. Declarations

7.1 Ethics Approval and Consent to Participate

This study does not involve human participants, human data, or animal subjects. Therefore, ethical approval and consent to participate are not applicable.

7.2 Consent for Publication

The author confirms that this manuscript does not contain any individual person's data in any form. Therefore, consent for publication is not applicable.

7.3 Availability of Data and Materials

The datasets generated and/or analyzed during the current study are available from the corresponding author upon reasonable request.

7.4 Conflicts of Interest

The author declares that there are no conflicts of interest regarding the publication of this paper.

7.5 Funding

This research received no external funding. The author declares that there are no financial or personal relationships that could have influenced the work reported in this study.

8. References

- [1] A. Agrawal, G. Dubey, M. Dubey, P. Narwaria, and D. Khatri, "NEXT GENERATION NETWORKS: ADVANCEMENTS, CHALLENGES, AND OPPORTUNITIES FOR SCALABLE AND SECURE INFRASTRUCTURE," 2024, pp. 58–71. DOI:[10.58532/V3BGNC2P2CH2](https://doi.org/10.58532/V3BGNC2P2CH2)
- [2] A. Kumar, M. Masud, M. H. Alsharif, N. Gaur, and A. Nanthamornphong, "Integrating 6G technology in smart hospitals: challenges and opportunities for enhanced healthcare services.," *Front. Med.*, vol. 12, p. 1534551, 2025. doi: [10.3389/fmed.2025.1534551](https://doi.org/10.3389/fmed.2025.1534551)
- [3] S. Katragadda and G. D., "Self-Healing Networks: Implementing AI-Powered Mechanisms to Automatically Detect and Resolve Network Issues with Minimal Human Intervention," Dec. 2023. <https://doi.org/10.3390/computers15040231>
- [4] N. Kumar, E. Groenewald, S. Kulkarni, and A. Kariveliparambil, "Self-Healing Networks AI-Based Approaches for Fault Detection and Recovery," *Power Syst. Technol.*, vol. 47, pp. 371–386, Dec. 2023. DOI:[10.52783/pst.206](https://doi.org/10.52783/pst.206)
- [5] G. Shingade, "Applications of Artificial Intelligence and Machine Learning in Power System Operation and Control: A Comprehensive Review," *Commun. Appl. Nonlinear Anal.*, vol. 32, pp. 2007–2022, Mar. 2025. DOI:[10.1109/TPWRS.2022.3220799](https://doi.org/10.1109/TPWRS.2022.3220799)
- [6] Y. Zahraoui *et al.*, "AI Applications to Enhance Resilience in Power Systems and Microgrids—A Review," *Sustainability*, vol. 16, no. 12, 2024. <https://doi.org/10.3390/su16124959>
- [7] A. Inamdar, W. D. van Driel, and G. Zhang, "Digital Twin Technology—A Review and Its Application Model for Prognostics and Health Management of Microelectronics," *Electronics*, vol. 13, no. 16, 2024. <https://doi.org/10.3390/electronics13163255>
- [8] A. Khoudi, T. Masrou, I. El Hassani, and C. El Mazgualdi, "A Deep-Reinforcement-Learning-Based Digital Twin for Manufacturing Process Optimization," *Systems*, vol. 12, no. 2, 2024. <https://doi.org/10.3390/systems12020038>
- [9] P. Almasan *et al.*, "Network Digital Twin : Context , Enabling Technologies and Opportunities," no. M1, pp. 1–7, 2022. DOI:[10.1109/MCOM.001.2200012](https://doi.org/10.1109/MCOM.001.2200012)
- [10] F. B. Mismar and B. L. Evans, "Deep Q-Learning for Self-Organizing Networks Fault Management and Radio Performance Improvement," 2019. DOI:[10.1109/ACSSC.2018.8645083](https://doi.org/10.1109/ACSSC.2018.8645083)
- [11] D. Saxena, A. K. Singh, and S. Member, "A Self-Healing and Fault-Tolerant Cloud-based," pp. 1–10, 2025. DOI:[10.48550/arXiv.2505.01215](https://doi.org/10.48550/arXiv.2505.01215)
- [12] N. Cheng, S. Member, X. W. Id, S. Member, Z. L. Id, and S. Member, "Toward Enhanced Reinforcement Learning-Based Resource Management via Digital Twin : Opportunities , Applications , and Challenges," pp. 1–7, 2024. DOI:[10.48550/arXiv.2505.01215](https://doi.org/10.48550/arXiv.2505.01215)
- [13] N. Majid and O. Isreal, "AI-Driven Network Automation: A Systematic Review of Self-Organizing and Self- Healing Computer Networks," Jan. 2026.
- [14] A. Hakiri, A. Gokhale, S. Ben Yahia, and N. Mellouli, "A comprehensive survey on digital twin for future networks and emerging Internet of Things industry," *Comput. Networks*, vol. 244, p. 110350, 2024. DOI:[10.1016/j.comnet.2024.110350](https://doi.org/10.1016/j.comnet.2024.110350)
- [15] T. Ward, N. T. City, K. H. Province, T. Engineering, and I. Technology, "Self-Healing Networks AI-Based Approaches for Fault Detection and Recovery. *1," vol. 47, no. 4, pp. 4–10, 2023. DOI:[10.52783/pst.206](https://doi.org/10.52783/pst.206)
- [16] Z. Noun, M. Saied, H. Muhieddine, H. Shraim, C. Francis, and H. Noura, "Reinforcement learning in fault tolerance and diagnosis fields: A literature review," *Annu. Rev. Control*, vol. 61, p. 101055, 2026. DOI:[10.1016/j.arcontrol.2026.101055](https://doi.org/10.1016/j.arcontrol.2026.101055)
- [17] R. van Dinter, B. Tekinerdogan, and C. Catal, "Predictive maintenance using digital twins: A systematic literature review," *Inf. Softw. Technol.*, vol. 151, p. 107008, 2022. <https://doi.org/10.1016/j.infsof.2022.107008>
- [18] Y. Yang, X. Chen, R. Tan, and Y. Xiao, "IoT Technologies and Applications," in *Intelligent IoT for the Digital World: Incorporating 5G Communications and Fog/Edge Computing Technologies*, 2021, pp. 1–60. DOI: [10.1002/9781119593584.ch7](https://doi.org/10.1002/9781119593584.ch7)
- [19] T. Omar, T. Ketseoglou, O. Naffaa, A. Marzvanyan, and C. Carr, "A Precoding Real-Time Buffer Based Self-Healing Solution for 5G Networks," *J. Comput. Commun.*, vol. 09, pp. 1–23, Jan. 2021. DOI:[10.3390/fi15070244](https://doi.org/10.3390/fi15070244)
- [20] S. A. Kauffman, A. Roli, A. Mater, and S. Università, "Beyond the Newtonian Paradigm : A Statistical Mechanics of Emergence," pp. 1–20, 2021. DOI:[10.1098/rsfs.2022.0063](https://doi.org/10.1098/rsfs.2022.0063)
- [21] A. Mujezinović, E. Turajlić, A. Alihodžić, M. M. Dedović, and N. Dautbašić, "Calculation of Magnetic Flux Density Harmonics in the Vicinity of Overhead Lines," *Electronics*, vol. 11, no. 4, 2022. <https://doi.org/10.3390/electronics11040512>
- [22] K. Madala, H. Do, and D. Aceituna, "A Combinatorial Approach for Exposing Off-Nominal Behaviors," in *2018 IEEE/ACM 40th International Conference on Software Engineering (ICSE)*, 2018, pp. 910–920. <https://doi.org/10.1145/3180155.3180204>

تصميم وتقييم شبكات قابلة للتعافي الذاتي باستخدام تعلم Q وتقنية التوأم الرقمي

ليث سامي عبد الكاظم

قسم هندسة الحاسوب والشبكات، جامعة الحكيم سبزواري، إيران

المستخلص:

تتطلب شبكات الاتصال الحديثة ذات التعقيد المتزايد اليوم خصائص أساسية لا غنى عنها في البنى التحتية المستقبلية، مثل الاعتمادية، والمرونة، وسرعة استعادة الأعطال. إلا أن حلول إدارة الأعطال الحالية تعتمد بشكل كبير على الأساليب اليدوية أو الأنظمة الثابتة المعتمدة على قواعد مسبقة، مما يجعلها غير قادرة على التعامل بكفاءة مع الأعطال الديناميكية والواسعة النطاق في الشبكات الحديثة. لذلك، يهدف هذا البحث إلى تصميم واختبار إطار عمل لشبكات ذاتية الإصلاح تعتمد على دمج التعلم المعزز (Reinforcement Learning) والتوأم الرقمي (Digital Twin) لتحقيق إصلاح ذاتي مستقل، تكيفي، وأني. تعتمد الفكرة الأساسية للنظام المقترح على تصميم وكيل تعلم باستخدام خوارزمية Q-learning يتفاعل باستمرار مع بيئة شبكة محاكاة لتعلم أفضل استراتيجيات الإصلاح، بالتوازي مع استخدام التوأم الرقمي لمحاكاة الشبكة الفعلية والتنبؤ بالأعطال المحتملة اعتماداً على البيانات التاريخية والبيانات اللحظية. يتكون الإطار المقترح من عدة وحدات معالجة، تشمل محاكي شبكة يدعم طوبولوجيات مختلفة مثل الشبكات صغيرة العالم، العشوائية، التكيفية، عديمة المقياس، الكاملة، النجمية، والشبكات الشبكية (Grid). كما يتضمن وحدة مراقبة وتنبؤ لحالة العقد والوصلات تعتمد على التوأم الرقمي، بالإضافة إلى خوارزمية Q-learning بأسلوب ϵ -greedy لاتخاذ قرارات تكيفية في عمليات الإصلاح، وآلية حساب درجة صحة مركبة تعتمد على حالة العقد والوصلات. كما يستخدم النظام نماذج للتنبؤ بالأعطال، وإدارة البيانات التاريخية، وتقنيات تحسين المكافأة لتعزيز تقارب التعلم وزيادة مرونة الشبكة. تشمل معايير التقييم: صحة الشبكة، متوسط زمن الإصلاح (MTTR)، دقة كشف الأعطال، معدل الإيجابيات الكاذبة، توافر الشبكة، ونسبة نجاح الإصلاح. أظهرت النتائج التجريبية أن إطار العمل المقترح (RL-Digital Twin) يتفوق بشكل ملحوظ على الطرق التقليدية مثل عدم القيام بأي إجراء، والإصلاح العشوائي، والإصلاح القائم على القواعد، حيث حقق متوسط صحة شبكة قدره 89.4%، ومتوسط زمن إصلاح يبلغ 3.1 خطوة، ودقة تنبؤ الأعطال وصلت إلى 91.2%. كما تمكن الوكيل الذكي من التكيف مع مختلف طوبولوجيات الشبكات وتحقيق تقارب خلال 65 إلى 100 حلقة تدريبية. ورغم هذه النتائج الإيجابية، فإن الدراسة تقتصر على شبكات محاكاة لا تتجاوز 30 عقدة، وتركز بشكل أساسي على الأعطال التشغيلية وليس الهجمات السيبرانية. كما أن هناك تحديات تتعلق بقابلية التوسع في الشبكات الأكبر بسبب القيود المرتبطة بخوارزمية Q-learning التقليدية.