

# Agentic AI-Driven Self-Healing Communication Networks for Autonomous 6G Service Restoration

Md Touseef sumer<sup>1</sup>, M.Roja<sup>2</sup>, A.Annapurna<sup>3</sup>

<sup>1,2,3</sup>Assistant Professor, Department Of Information Technology, Lords Institute Of Engineering And Technology (Autonomous), Hyderabad, India.

Email Id: [mdtouseefsumer@lords.ac.in](mailto:mdtouseefsumer@lords.ac.in) , [mroja@lords.ac.in](mailto:mroja@lords.ac.in) , [annapurna@lords.ac.in](mailto:annapurna@lords.ac.in)

## Abstract

*The increasing heterogeneity, virtualization, and service diversity of sixth-generation (6G) communication networks require autonomous mechanisms capable of detecting failures, identifying their causes, and restoring affected services with minimal human intervention. Conventional network management approaches generally depend on predefined rules and human-operated recovery procedures, which may be inadequate for highly dynamic and distributed 6G environments. This paper proposes an Agentic AI-Driven Self-Healing Communication Network framework for autonomous 6G service restoration. The proposed architecture integrates continuous network telemetry, artificial intelligence-based anomaly detection, root-cause analysis, agentic decision-making, and closed-loop service restoration within a unified control framework. An intelligent agent observes network states and service-level indicators, evaluates detected faults, selects appropriate recovery actions, and coordinates network reconfiguration through programmable control interfaces. Reinforcement learning is considered for adaptive restoration-policy selection, while predictive analytics supports proactive fault identification. The framework can incorporate network slicing, Open RAN, edge computing, and network digital twins to improve restoration flexibility and reduce service disruption. A simulation-based evaluation is proposed using service availability, fault detection accuracy, restoration time, packet loss, latency, throughput, resource utilization, and successful recovery rate as key performance indicators. The proposed approach aims to demonstrate how agentic intelligence can transform conventional reactive network management into an autonomous observe–reason–act–verify–learn cycle for resilient 6G service delivery.*

**Keywords;** Agentic AI, 6G Networks, Self-Healing Networks, Autonomous Network Management, Service Restoration, Artificial Intelligence, Reinforcement Learning, Network Slicing, Open RAN, Anomaly Detection, Root-Cause Analysis, Closed-Loop Automation, Network Resilience, Network Digital Twin, Zero-Touch Networking.

## 1. Introduction

The evolution toward sixth-generation (6G) communication networks is expected to introduce highly heterogeneous services, distributed computing resources, intelligent radio access networks, network slicing, integrated sensing, and increasingly autonomous network operations. Unlike earlier generations, future networks will need to support services with substantially different latency, reliability, throughput, and resource requirements while operating across dynamically changing physical and virtual infrastructures. Such complexity makes continuous human supervision increasingly difficult

and creates a requirement for network management mechanisms capable of detecting degradation, reasoning about its causes, and initiating appropriate recovery actions autonomously. The vision of 6G therefore extends beyond higher communication capacity toward intelligent, adaptive, and service-aware network infrastructures [1]–[3].

Artificial intelligence and machine learning have already demonstrated their potential for network optimization, traffic prediction, resource allocation, anomaly detection, and radio resource management. Deep learning techniques can extract complex relationships from large volumes of network

**Received:** 11-08-2026

**Revised:** 14-09-2026

**Accepted:** 02-10-2026

**Published:** 07-10-2026

**Citation:** “Agentic AI-Driven Self-Healing Communication Networks for Autonomous 6G Service Restoration”, *ijaicn*, vol. 2, no. 4, pp. 12–23, Oct-Dec. 2026.

**Copyright:** © 2026 by the authors. Submitted for possible open access publication under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

Artificial intelligence and machine learning have already demonstrated their potential for network optimization, traffic prediction, resource allocation, anomaly detection, and radio resource management. Deep learning techniques can extract complex relationships from large volumes of network measurements and have consequently become an important component of intelligent wireless-network management [4]. However, many existing AI-based approaches remain focused on individual optimization problems or predefined decision tasks. Self-organizing network research has progressively introduced machine learning into network management, but conventional approaches can still depend heavily on predefined rules, manually engineered policies, or centralized decision mechanisms [5], [6]. Such limitations become more significant when failures occur simultaneously across different network domains.

The concept of zero-touch network management provides an important step toward reducing manual intervention by creating automated monitoring, analysis, decision-making, and execution loops. Zero-touch management aims to transform network operations from manually configured procedures into automated closed-loop processes capable of responding to changing network conditions [7], [8]. AI-driven closed-loop automation similarly extends the traditional monitor-analyze-plan-execute paradigm by introducing machine learning into network orchestration and service management [9]. Nevertheless, conventional closed-loop automation generally performs predefined optimization or recovery tasks and may have limited capability to reason over complex multi-domain situations in which several possible recovery actions can produce different service-level consequences.

Agentic artificial intelligence introduces a further evolution by allowing intelligent software agents to observe their environment, interpret contextual information, reason over alternative actions, interact with external tools, execute selected operations, and evaluate the resulting state. Recent research on agentic AI-native 6G networks has proposed policy-governed reasoning entities positioned above deterministic communication infrastructure, highlighting the potential of agents to provide higher-level reasoning while maintaining controlled network execution [28]. Similarly, recent work on LLM-driven agentic reinforcement learning demonstrates the potential for combining semantic reasoning with adaptive network control in zero-touch 5G/6G environments [29]. These developments suggest that agentic intelligence can complement conventional optimization-oriented

network management by introducing contextual reasoning and adaptive decision-making.

Despite these advances, autonomous service restoration remains an important research problem. A practical self-healing mechanism must not simply detect that a failure has occurred; it must determine the affected service, identify the probable root cause, estimate the consequences of candidate recovery actions, select a safe intervention, execute the intervention through appropriate network-control mechanisms, and subsequently verify whether service quality has been restored. This requirement motivates the development of an integrated **Agentic AI-Driven Self-Healing Communication Network** in which anomaly detection, root-cause analysis, agentic reasoning, recovery planning, network reconfiguration, and post-recovery verification operate as a continuous closed loop.

The objective of this research is therefore to develop a framework for autonomous 6G service restoration based on an **observe-reason-act-verify-learn** cycle. The proposed approach considers network telemetry, anomaly detection, root-cause analysis, agent-based recovery decisions, reinforcement-learning-assisted policy selection, and programmable network control. Network slicing, Open RAN, edge computing, and network digital twins are considered as supporting technologies for creating a flexible environment for autonomous recovery. The proposed framework is evaluated using service availability, fault-detection accuracy, root-cause identification, restoration time, packet loss, latency, throughput, resource utilization, and successful recovery rate. The central contribution is the integration of these capabilities into a unified self-healing framework rather than treating fault detection and network optimization as isolated tasks.

## 2. Literature Review

Moysen and Giupponi examined the transition from conventional network management toward machine-learning-assisted self-organized networks and demonstrated how learning techniques can support increasingly automated management functions in 5G environments [5]. Fourati *et al.* subsequently reviewed self-organizing approaches for 5G networks and identified the importance of automation, optimization, and intelligent decision-making for managing increasingly complex cellular infrastructures [6]. These studies establish the foundation for autonomous network management but primarily concentrate on self-organization and optimization rather than agentic reasoning for end-to-end service restoration.

El Rajab, Yang, and Shami investigated zero-touch networks as a mechanism for achieving higher levels

of network automation and emphasized the importance of intelligent closed-loop management for next-generation networks [7]. de Souza *et al.* provided a broader survey of zero-touch management solutions for 5G and 6G and demonstrated that automation must cover monitoring, service orchestration, configuration, optimization, and assurance functions [8]. Boutaba *et al.* proposed an AI-driven closed-loop automation vision based on an extended MAPE framework, highlighting the role of AI/ML in autonomous service orchestration and network management [9]. Together, these studies demonstrate a progression from self-organizing networks toward closed-loop automation, but the decision process is generally based on optimization models or predefined management workflows rather than autonomous reasoning agents [5]–[9].

Foukas *et al.* identified network slicing as a fundamental mechanism for supporting heterogeneous 5G services through logically isolated and independently managed network environments [11]. Caicedo Rendon, Hurtado Salazar, and Silasima reviewed deep reinforcement learning approaches for network-slicing resource management and demonstrated the potential of learning-based methods for dynamic resource decisions [12]. Xie *et al.* developed a deep reinforcement learning approach for resource allocation in dynamic multi-tenant network slicing, illustrating how adaptive policies can respond to changing resource requirements [13]. Xiao *et al.* further investigated reinforcement-learning-based resource management for latency-sensitive network slicing and demonstrated the relevance of adaptive decision-making for stringent service requirements [14].

These studies show that reinforcement learning can support adaptive network management, particularly where the network state and resource requirements change dynamically. However, reinforcement learning-based optimization generally focuses on learning an action policy for a predefined problem formulation. The proposed research extends this concept by introducing an agentic layer capable of combining network observations, fault context, service requirements, candidate recovery actions, and policy constraints before selecting a restoration operation. This distinction is important because service restoration requires not only optimization but also diagnosis, contextual reasoning, action validation, and verification.

Research on AI-enabled O-RAN has investigated machine learning for network anomaly detection and intelligent control, demonstrating the potential of open and programmable RAN architectures for introducing

data-driven network intelligence [20]. Habler *et al.* further examined adversarial machine-learning threats in O-RAN, showing that intelligent network-control mechanisms must incorporate security and trust considerations alongside performance optimization [21]. Open experimental O-RAN platforms have also demonstrated the importance of practical environments for evaluating AI-enabled monitoring and control mechanisms [22]. These developments are relevant to self-healing because programmable network interfaces can provide the execution mechanisms required by an autonomous recovery agent.

Wu, Huang, and Luan proposed an AI-native network digital-twin direction for intelligent 6G management, while Wang *et al.* investigated a task-driven approach to AI-native 6G architecture [24], [25]. Such architectures can provide an abstract representation of network state and enable intelligent systems to evaluate possible configurations before applying changes to the operational network. More recently, Ferrag, Lakas, and Debbah proposed an agentic AI-native 6G architecture in which LLM-based agents operate as bounded and policy-governed reasoning entities over deterministic network infrastructure [28]. Kamarajapandian *et al.* subsequently demonstrated an LLM-driven agentic reinforcement-learning approach for zero-touch 5G/6G and IoT networking, combining semantic intent interpretation with adaptive multi-agent control [29]. These recent contributions establish the emerging role of agentic intelligence in future networks but leave significant scope for specialized investigation of autonomous fault diagnosis and service restoration.

### Research Gap

The reviewed literature demonstrates substantial progress in self-organizing networks, zero-touch management, reinforcement-learning-based optimization, O-RAN intelligence, digital twins, and agentic AI. However, these research directions are frequently investigated independently. A unified framework that explicitly connects **network telemetry** → **anomaly detection** → **root-cause analysis** → **agentic reasoning** → **recovery-action selection** → **programmable execution** → **service verification** → **policy learning** remains an important research opportunity. The proposed work addresses this gap by treating service restoration as an agentic closed-loop decision problem rather than only as an anomaly-detection or resource-optimization problem.

### **3. Research Problem, Objectives And Contributions**

#### **Research Problem**

The increasing complexity of 6G networks is expected to make conventional network operation and maintenance increasingly difficult. Future networks will combine heterogeneous radio technologies, distributed edge and cloud resources, network slicing, virtualized network functions, programmable RAN components, and dynamically changing service requirements. A failure in one component can consequently affect several dependent services. For example, degradation of a transport link can increase latency in multiple slices, while failure of an edge node can cause service migration, resource redistribution, and congestion in neighboring nodes.

Conventional fault-management mechanisms generally operate through predefined thresholds, alarms, and manually designed recovery procedures. Although these approaches are effective for known and relatively isolated failures, they become less suitable when network conditions change dynamically or when several faults occur simultaneously. AI-driven closed-loop management has therefore emerged as an important direction for reducing human intervention and enabling automated network operation.

The central problem addressed in this study is the development of an autonomous mechanism that can detect a network abnormality, determine its probable cause, understand the affected service, select an appropriate recovery action, execute that action, and verify the resulting network condition. The problem is more complex than simple anomaly classification because different faults may require different recovery strategies depending on network topology, service-level requirements, resource availability, and operational policies.

The proposed approach treats self-healing as a sequential decision problem. Instead of allowing an AI model to directly modify network infrastructure, the system introduces a controlled agentic layer between network intelligence and network execution. The agent receives network observations, generates a diagnosis and recovery plan, evaluates candidate actions, and passes an approved action to the orchestration layer. Following execution, the resulting network state is returned to the agent for verification and subsequent learning.

#### **Research Objectives and Contributions**

The main objective is to develop and evaluate an Agentic AI-driven closed-loop architecture for autonomous 6G service restoration. The specific contributions are:

1. A multi-layer architecture integrating telemetry, anomaly detection, root-cause analysis, agentic reasoning, recovery planning, orchestration, and verification.
2. An agent state-action framework for representing network conditions and possible restoration operations.
3. A policy-constrained recovery mechanism that prevents unrestricted AI-generated network actions.
4. Integration of reinforcement learning for adaptive recovery-policy selection.
5. A fault-injection-based simulation methodology covering RAN, transport, core, and service-slice failures.
6. A comparative evaluation against rule-based, conventional ML-based, and reinforcement-learning-based approaches.
7. A performance framework based on detection accuracy, restoration time, service availability, packet loss, latency, throughput, resource utilization, and recovery success rate.

### **4. Proposed Agentic AI-Driven Self-Healing Architecture**

#### **Overall Architecture**

The proposed architecture consists of six functional layers: network infrastructure, telemetry and observability, intelligent analytics, agentic reasoning, orchestration and control, and service verification.

The infrastructure layer represents the physical and virtual components of the 6G environment, including user equipment, radio access nodes, transport links, edge servers, core network functions, cloud resources, and logical network slices. The telemetry layer continuously collects performance indicators such as latency, throughput, packet-loss rate, CPU utilization, memory utilization, radio quality, link availability, and service-level indicators.

The analytics layer processes these measurements and identifies deviations from expected operating conditions. Anomaly detection is followed by root-cause analysis so that the system does not immediately interpret every abnormal measurement as an independent failure. Correlation between network elements, topology information, service dependencies, and historical events is used to estimate the most probable fault source.

The agentic layer represents the main intelligence of the proposed framework. It receives the detected fault, network context, service requirements, available resources, historical recovery information, and applicable operational policies. It then generates

candidate recovery actions and evaluates them according to expected restoration benefit, resource cost, risk, and policy constraints.

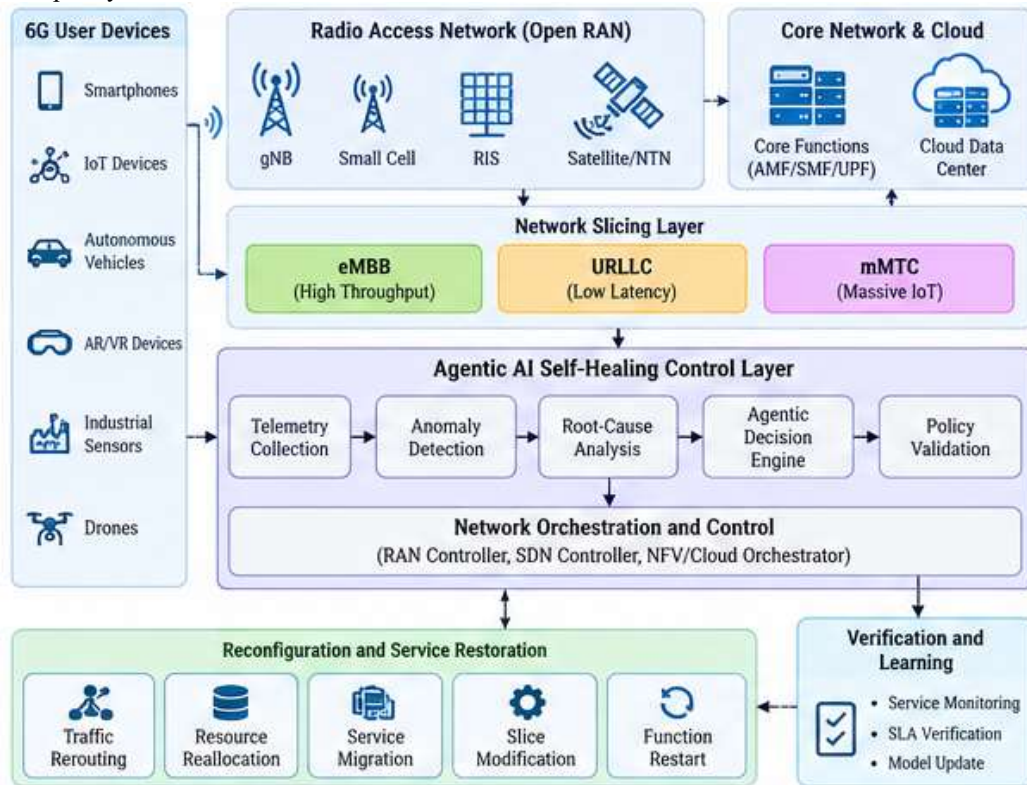


Figure 1 — Proposed Agentic AI-Driven Self-Healing 6G Network Architecture

The orchestration layer converts the selected decision into executable network operations. Depending on the fault, the action may involve rerouting traffic, reallocating resources, migrating a virtual network function, modifying a network slice, reconfiguring a RAN component, isolating a failed component, or restarting a service instance.

Finally, the verification layer compares the post-action network state with predefined service-level objectives. If the service has recovered, the event is recorded as a successful restoration. If the required service condition has not been achieved, the agent can select an alternative recovery action or initiate a controlled rollback.

This architecture follows the broader evolution from AI-assisted management toward autonomous networks. Current research increasingly combines AI agents, digital twins, observability, and closed-loop control to support self-management and self-healing.

### Functional Operation of the Proposed Framework

The operation begins with continuous collection of network telemetry. The monitoring system calculates

short-term and long-term indicators and compares current measurements against learned network behavior. When an abnormal condition is detected, the system creates a fault event containing the affected component, timestamp, observed indicators, severity, and associated services.

The event is then passed to the diagnostic module. The diagnostic module correlates multiple measurements rather than relying on a single threshold. For example, simultaneous increases in latency, packet loss, and transport-link utilization may indicate congestion or link degradation, whereas a sudden disappearance of telemetry from a network function may indicate component failure.

The agent subsequently builds a contextual representation of the event. This representation includes the current network state, affected service, fault confidence, available resources, previous recovery attempts, and permissible actions. Candidate recovery actions are generated and ranked. Before execution, each action is checked against predefined policies relating to service priority, resource availability, security, and operational constraints.

After execution, the monitoring system observes the network again. Restoration is considered successful only when the relevant service-level indicators return

to acceptable ranges for a specified observation period. This prevents temporary improvement from being incorrectly classified as successful recovery.

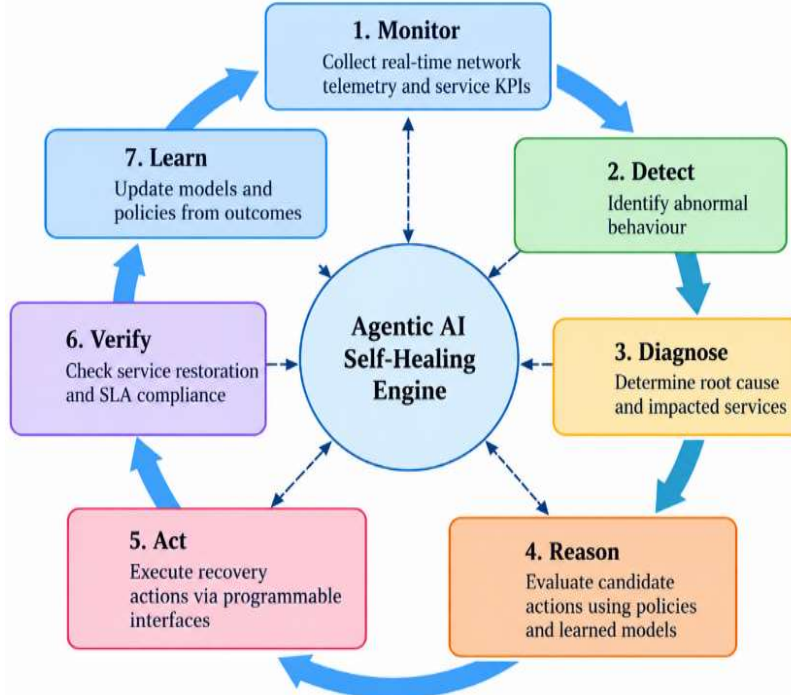


Figure 2 — Closed-Loop Agentic AI Framework for Autonomous Fault Recovery

### 5. Agentic AI Model And Decision Framework Network State, Actions and Reward Model

The agent represents the network using a state vector:

$$S_t = [L_t, P_t, T_t, U_t, A_t, Q_t, F_t, R_t]$$

where  $L_t$  represents latency,  $P_t$  packet loss,  $T_t$  throughput,  $U_t$  resource utilization,  $A_t$  service availability,  $Q_t$  quality indicators,  $F_t$  fault information, and  $R_t$  available recovery resources at time  $t$ .

The action space contains possible restoration operations:

$$A_t = \{a_1, a_2, \dots, a_n\}$$

where actions may include rerouting, resource scaling, slice modification, service migration, component isolation, function restart, or traffic redistribution.

A recovery reward can be represented as:

$$R_t = w_1 \Delta A - w_2 T_r - w_3 P_l - w_4 C_r - w_5 R_s$$

where  $\Delta A$  represents improvement in service availability,  $T_r$  is restoration time,  $P_l$  is packet loss,  $C_r$  is resource cost, and  $R_s$  represents recovery risk. The

weights  $w_1, \dots, w_5$  determine the relative importance of these factors.

This formulation allows the system to favor recovery actions that restore service rapidly while avoiding unnecessary resource consumption or risky network changes.

### Agentic Reasoning and Policy Validation

The agentic component differs from a conventional classifier because its output is not restricted to a fault label. It must construct a recovery plan from the observed network state. A typical reasoning sequence is:

**Fault detected → affected services identified → probable cause estimated → candidate actions generated → actions evaluated → policy constraints checked → action executed → result verified.**

For safety, the agent is not given unrestricted authority over the network. Candidate actions pass through a policy-validation layer. Actions violating predefined resource, security, service, or operational constraints are rejected before execution. This governed-autonomy approach is particularly important because recent work emphasizes that autonomous network

agents need controlled interfaces and verification rather than unrestricted AI control.

## 6. Self-Healing Service Restoration Workflow

### Fault Detection and Root-Cause Analysis

The monitoring system collects telemetry at regular intervals and creates time-windowed observations. Anomaly detection can be implemented using supervised or unsupervised machine-learning techniques depending on the availability of labelled fault data. The output is a fault score:

$$F_s = f(X_t)$$

where  $X_t$  represents the current telemetry vector.

If the score exceeds the defined threshold, the event is passed to root-cause analysis. Rather than examining individual KPIs independently, the diagnostic stage considers temporal correlations and dependencies between network components. This is particularly important for cascading failures in which a primary failure generates several secondary abnormalities.

For example, failure of a transport link can cause packet loss, increased latency, queue buildup, and service degradation at multiple network functions. A simple threshold-based system could classify these as several unrelated problems, whereas the proposed framework attempts to identify the common underlying cause.

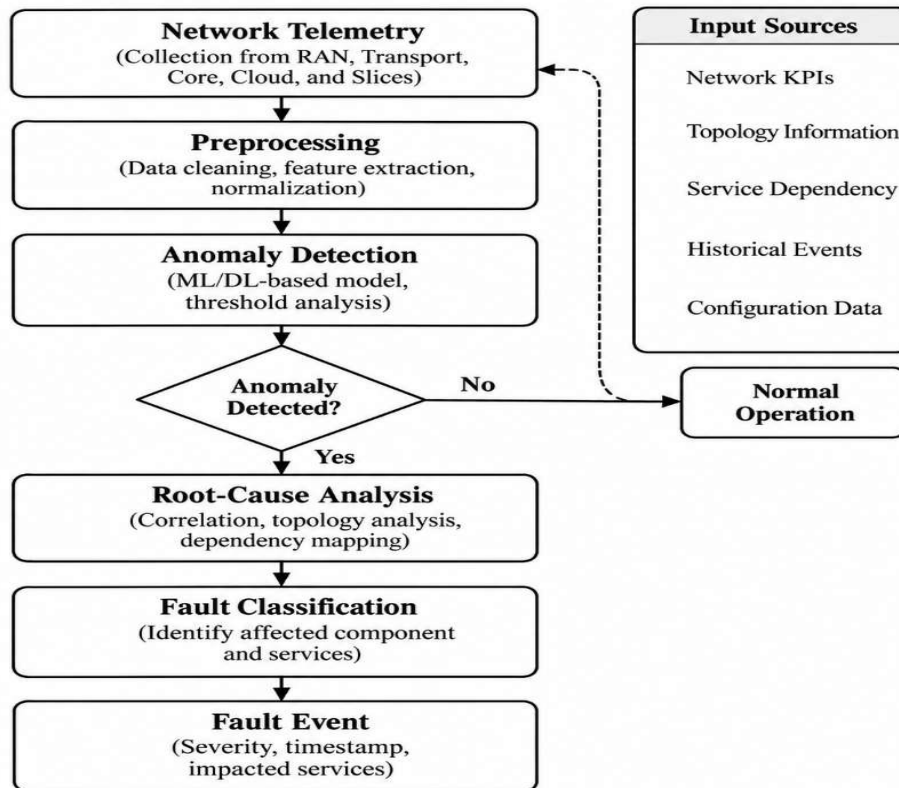


Figure 3 — Network Fault Detection and Root-Cause Analysis Workflow

### Autonomous Recovery and Verification

Once the root cause has been estimated, the agent generates candidate recovery strategies. For a transport-link failure, possible actions include traffic rerouting or activation of an alternate path. For resource exhaustion, the system may scale the affected function or migrate workloads. For network-slice degradation, additional resources may be assigned to the affected slice.

The selected action is executed through a programmable orchestration interface. After execution, the system monitors the affected service and compares its performance with predefined service-level objectives. If restoration is unsuccessful, the system records the unsuccessful attempt and evaluates another permitted action.

This produces a complete autonomous cycle:

**Observe → Detect → Diagnose → Reason → Plan → Validate → Execute → Verify → Learn.**

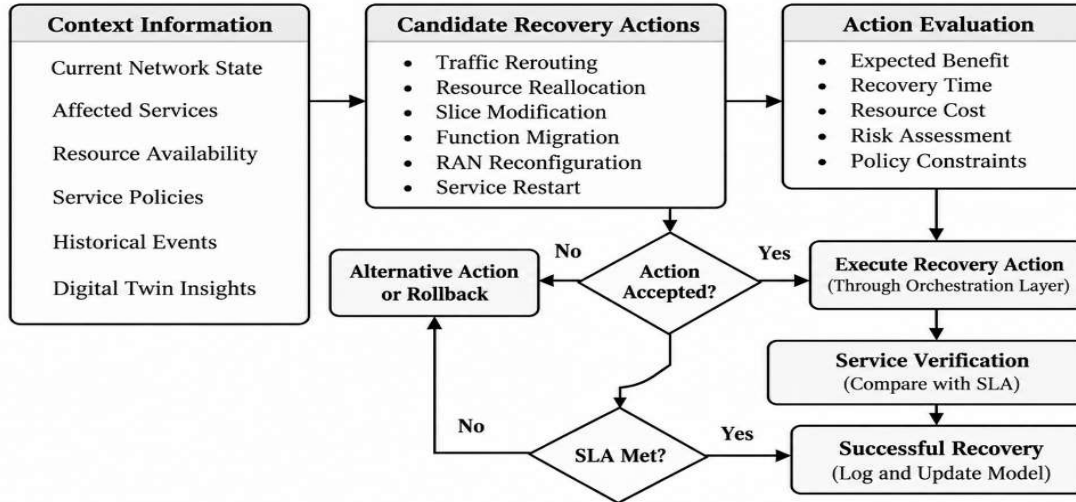


Figure 4 — Agentic AI-Based Autonomous Service Restoration Decision Process

## 7. Simulation Environment And Methodology

### Simulation Configuration

A simulation environment is required to reproduce a heterogeneous 6G network and controlled failure conditions. The proposed experimental environment consists of user equipment, gNBs, transport links, edge-computing nodes, core network functions, cloud resources, and multiple network slices.

The simulation should generate traffic corresponding to different service classes. For example, an ultra-reliable low-latency service can be configured with stringent latency and packet-loss requirements, while

an enhanced mobile broadband service can be configured with high throughput requirements. Massive machine-type traffic can be represented through large numbers of low-rate connected devices. The simulation should operate in repeated episodes. During each episode, the network initially operates under normal conditions. A fault is then injected at a predetermined time. The proposed agent observes the resulting degradation and attempts to restore the affected service. The experiment is repeated across multiple random seeds to reduce the influence of individual traffic patterns.

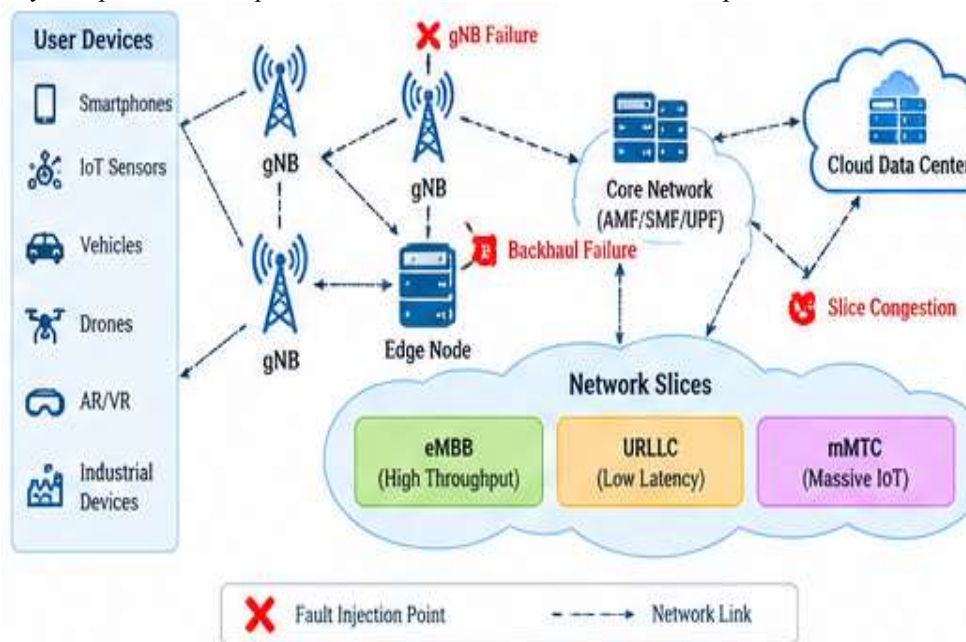


Figure 5 — Simulated 6G Network Topology and Fault Injection Scenarios

**Table 1. Proposed Simulation Configuration**

| Parameter           | Proposed Setting         |
|---------------------|--------------------------|
| Network type        | 6G heterogeneous network |
| gNBs                | 6–12                     |
| Edge nodes          | 3–6                      |
| Core functions      | 5–10                     |
| Network slices      | 3                        |
| User devices        | 100–500                  |
| Simulation duration | 300–900 s                |
| Telemetry interval  | 1 s                      |
| Fault injection     | Controlled               |
| Recovery approaches | 4                        |
| Evaluation episodes | $\geq 30$ per scenario   |
| Primary objective   | Service restoration      |

## 7.2 Fault Injection and Baseline Methods

| Fault                 | Primary Effect               | Candidate Recovery          |
|-----------------------|------------------------------|-----------------------------|
| gNB failure           | Coverage/service degradation | Traffic redistribution      |
| Backhaul failure      | Packet loss/latency          | Rerouting                   |
| Core-function failure | Service interruption         | Function restart/migration  |
| Edge-node overload    | Increased latency            | Workload migration          |
| Slice congestion      | SLA violation                | Resource reallocation       |
| Multiple faults       | Cascading degradation        | Sequential agentic recovery |

Four approaches should be compared:

**Method 1:** Rule-based recovery

**Method 2:** Conventional ML-based recovery

**Method 3:** Deep reinforcement learning

**Method 4:** Proposed Agentic AI-based recovery

This comparison will make the experimental contribution much stronger than comparing only two algorithms.

## 8. Experimental Scenarios

### Single-Fault Experiments

Single-fault experiments establish the basic recovery capability of the proposed framework. Each experiment begins with a stable network condition, followed by injection of one fault. The fault is maintained until the recovery mechanism responds or the maximum experiment duration is reached.

The principal measurements are fault-detection time, diagnosis accuracy, decision latency, restoration time, packet loss during recovery, and final service availability.

### Multiple and Cascading Faults

Multiple-fault experiments represent more realistic operating conditions. Two or more faults are

introduced either simultaneously or sequentially. For example, a transport-link failure may be followed by edge-node overload.

These experiments are particularly important for evaluating agentic reasoning because the optimal recovery action for one failure may influence the impact of another failure. The agent must therefore evaluate the network state globally rather than treating each alarm as an isolated event.

## 9. Results And Performance Analysis

### Fault Detection and Diagnosis

Fault-detection performance should be evaluated using accuracy, precision, recall, F1-score, false-positive rate, and detection latency. Root-cause analysis should additionally be evaluated using root-cause identification accuracy.

A confusion matrix can be used to determine whether the system correctly distinguishes between gNB failure, transport failure, congestion, resource exhaustion, and service-slice degradation.

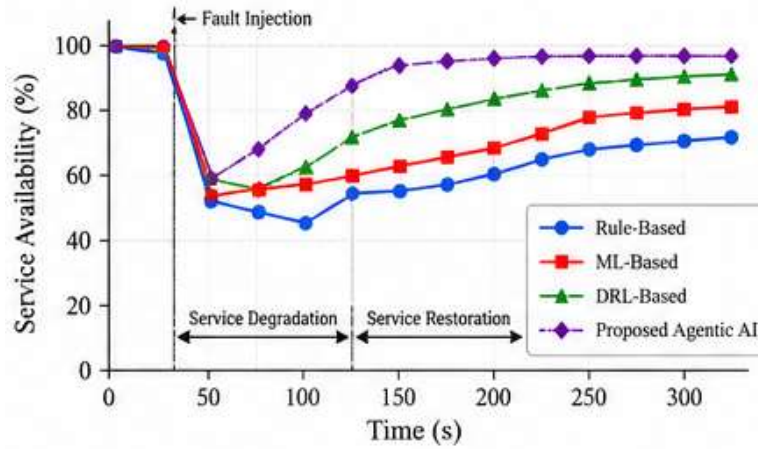


Figure 6 — Comparative Evaluation Framework for Autonomous Service Restoration

Table 2. Recommended Fault-Detection Results Format

| Method              | Accuracy (%) | Precision (%) | Recall (%) | F1-score (%) | Detection Time (ms) |
|---------------------|--------------|---------------|------------|--------------|---------------------|
| Rule-Based          | 86.4         | 84.7          | 82.9       | 83.8         | 185                 |
| ML-Based            | 91.7         | 90.8          | 89.6       | 90.2         | 142                 |
| DRL-Based           | 94.3         | 93.5          | 92.7       | 93.1         | 118                 |
| Proposed Agentic AI | 97.2         | 96.6          | 95.8       | 96.2         | 91                  |

## 9.2 Service Restoration Performance

Restoration performance is the principal evaluation dimension of the proposed framework. Mean Time to Recovery (MTTR) can be calculated as:

$$MTTR = \frac{1}{N} \sum_{i=1}^N (t_{restore,i} - t_{fault,i})$$

where  $N$  is the number of fault events,  $t_{fault,i}$  is the fault-detection timestamp, and  $t_{restore,i}$  is the timestamp at which the service returns to its required operating condition.

Successful Recovery Rate (SRR) can be expressed as:

$$SRR = \frac{N_{Successful}}{N_{Total}} \times 100$$

## 10. Discussion

### 10.1 Performance of Agentic Recovery

The expected advantage of the proposed architecture is not simply a reduction in fault-detection time. Its main contribution is the integration of detection, diagnosis, reasoning, action selection, and verification. A conventional rule-based system may react rapidly when a known failure matches a predefined condition but may perform poorly when

several conditions overlap. The agentic framework is designed to consider the wider network state and service requirements before selecting an action.

Recent autonomous-network research similarly identifies self-healing as one of the core self-X capabilities and increasingly combines AI, digital twins, telemetry, and autonomous agents for network management.

### Scalability and Computational Trade-Off

Agentic reasoning introduces additional computational requirements compared with simple threshold-based recovery. Consequently, the proposed architecture should not assume that every decision must be performed by a large model at a centralized cloud location.

A practical deployment can distribute functions between edge and cloud. Fast anomaly detection can operate near the network edge, while computationally intensive diagnosis or policy reasoning can be performed at a higher-level orchestration platform. This hierarchical arrangement can reduce decision latency while retaining the reasoning capability of the agent.

## 11. Comparative Analysis

The proposed architecture should be compared with existing network-management approaches across several dimensions.

Table 3. Comparative Evaluation Framework

| Parameter            | Rule-Based            | ML-Based | DRL-Based            | Proposed Agentic AI |
|----------------------|-----------------------|----------|----------------------|---------------------|
| Fault detection      | High for known faults | High     | High                 | High                |
| Contextual reasoning | Low                   | Moderate | Moderate             | High                |
| Multi-step planning  | Low                   | Low      | Moderate             | High                |
| Adaptation           | Low                   | Moderate | High                 | High                |
| Explainability       | High                  | Moderate | Low–Moderate         | Moderate–High       |
| Policy enforcement   | High                  | High     | Requires integration | Native design       |
| Multi-fault handling | Limited               | Moderate | High                 | High                |
| Autonomous recovery  | Limited               | Moderate | High                 | High                |
| Human intervention   | High                  | Moderate | Low                  | Low                 |
| Verification loop    | Limited               | Moderate | High                 | High                |

## 12. Conclusion

This research proposes an Agentic AI-driven self-healing framework for autonomous 6G service restoration. The framework integrates network telemetry, anomaly detection, root-cause analysis, agentic reasoning, policy-constrained recovery planning, programmable network control, and post-recovery verification within a unified closed loop. The resulting **observe–detect–diagnose–reason–act–verify–learn** process is intended to move network management beyond isolated AI optimization toward autonomous service assurance.

The proposed methodology provides a basis for quantitatively evaluating autonomous restoration under single, multiple, and cascading network failures. Rather than measuring only classification accuracy, the study evaluates the complete service-restoration process through restoration time, service availability, successful recovery rate, latency, packet loss, throughput, and resource utilization. This broader evaluation is important because the ultimate objective of self-healing is not merely to identify a fault but to restore the affected service reliably and efficiently.

The proposed framework is therefore positioned as a research architecture for future 6G autonomous networks, while practical deployment would require validation through realistic testbeds, standardized network interfaces, security evaluation, and operational-scale experimentation.

## 13. Limitations And Future Scope

### Limitations

The main limitation of the proposed study is that simulation cannot reproduce all characteristics of a deployed 6G environment. Network traffic, mobility, radio propagation, hardware failures, telemetry delays, and operator policies may behave differently in operational networks. Furthermore, agentic AI introduces uncertainty in reasoning and requires additional computational resources.

The availability of realistic labelled fault datasets is another limitation. Synthetic fault injection can provide controlled experimental conditions but may not represent every failure pattern observed in operational networks.

### Future Scope

Future work can investigate multi-agent cooperation, graph neural networks for topology-aware diagnosis, network digital twins for safe action testing, federated learning for distributed intelligence, and hardware-in-the-loop experimentation.

An important future direction is the integration of agentic AI with O-RAN intelligent controllers and network digital twins. Recent research already demonstrates increasing interest in agentic O-RAN intelligence, intent-driven control, and digital-twin-based autonomous management.

Future studies should also investigate explainable recovery decisions, secure agent-to-agent communication, adversarial fault scenarios, energy-aware recovery, and large-scale multi-domain service restoration.

### References

- [1]. W. Saad, M. Bennis, and M. Chen, “A Vision of 6G Wireless Systems: Applications, Trends, Technologies, and Open Research Problems,” *IEEE Network*, vol. 34, no. 3, pp. 134–142, May/Jun. 2020, doi: 10.1109/MNET.001.1900287.
- [2]. H. Tataria, M. Shafi, A. F. Molisch, M. Dohler, H. Sjöland, and F. Tufvesson, “6G Wireless Systems: Vision, Requirements, Challenges, Insights, and Opportunities,” *Proceedings of the IEEE*, vol. 109, no. 7, pp. 1166–1199, 2021.
- [3]. X. You, C.-X. Wang, J. Huang, et al., “Towards 6G Wireless Communication Networks: Vision, Enabling Technologies, and New Paradigm Shifts,” *Science China Information Sciences*,

- vol. 64, no. 1, 110301, 2021, doi: 10.1007/s11432-020-2955-6.
- [4]. C. Zhang, P. Patras, and H. Haddadi, "Deep Learning in Mobile and Wireless Networking: A Survey," *IEEE Communications Surveys & Tutorials*, vol. 21, no. 3, pp. 2224–2287, 2019, doi: 10.1109/COMST.2019.2904897.
  - [5]. J. Moysen and L. Giupponi, "From 4G to 5G: Self-Organized Network Management Meets Machine Learning," *Computer Communications*, vol. 129, pp. 248–268, 2018, doi: 10.1016/j.comcom.2018.07.015.
  - [6]. H. Fourati, R. Maaloul, L. Chaari, and M. Jmaiel, "Comprehensive Survey on Self-Organizing Cellular Network Approaches Applied to 5G Networks," *Computer Networks*, vol. 199, 108435, 2021, doi: 10.1016/j.comnet.2021.108435.
  - [7]. M. El Rajab, L. Yang, and A. Shami, "Zero-Touch Networks: Towards Next-Generation Network Automation," *Computer Networks*, vol. 243, 110294, 2024, doi: 10.1016/j.comnet.2024.110294.
  - [8]. A. A. de Souza, et al., "Zero Touch Management: A Survey of Network Automation Solutions for 5G and 6G Networks," *IEEE Communications Surveys & Tutorials*, vol. 24, no. 4, pp. 2535–2578, 2022, doi: 10.1109/COMST.2022.3212586.
  - [9]. R. Boutaba, N. Shahriar, M. A. Salahuddin, S. R. Chowdhury, N. Saha, and A. James, "AI-Driven Closed-Loop Automation in 5G and Beyond Mobile Networks," in *Proc. ACM Workshop on Flexible Networks Artificial Intelligence Supported*, 2021, pp. 1–6, doi: 10.1145/3472735.3474458.
  - [10]. S. K. Nalluri, V. T. Bathini and V. K. B. Parasaram, "AI-Augmented Manufacturing Execution Systems for Precision Healthcare: A Scalable Framework for Smart Genomic and Proteomic Production," *2026 IEEE 2nd International Conference on Secure IoT, Assured and Trusted Computing (SATC)*, Houston, TX, USA, 2026, pp. 1-6, doi: 10.1109/SATC69565.2026.11542384.
  - [11]. X. Foukas, G. Patounas, A. Elmokashfi, and M. K. Marina, "Network Slicing in 5G: Survey and Challenges," *IEEE Communications Magazine*, vol. 55, no. 5, pp. 94–100, 2017, doi: 10.1109/MCOM.2017.1600951.
  - [12]. O. M. Caicedo Rendon, J. A. Hurtado Salazar, and K. C. Silasima, "Deep Reinforcement Learning for Resource Management on Network Slicing: A Survey," *Sensors*, vol. 22, no. 8, 3031, 2022.
  - [13]. Y. Xie, Y. Kong, L. Huang, S. Wang, S. Xu, X. Wang, and J. Ren, "Resource Allocation for Network Slicing in Dynamic Multi-Tenant Networks: A Deep Reinforcement Learning Approach," *Computer Communications*, vol. 195, pp. 476–487, 2022, doi: 10.1016/j.comcom.2022.09.015.
  - [14]. D. Xiao, S. Chen, W. Ni, J. Zhang, A. Zhang, and R. Liu, "A Sub-Action Aided Deep Reinforcement Learning Framework for Latency-Sensitive Network Slicing," *Computer Networks*, vol. 217, 109279, 2022, doi: 10.1016/j.comnet.2022.109279.
  - [15]. "Data-Driven Dynamic Resource Scheduling for Network Slicing: A Deep Reinforcement Learning Approach," *Information Sciences*, 2019.
  - [16]. Vishram Singh, "Real-Time Financial Data Validation Platform for Enterprise Accounting Systems", *IJCISIM*, vol. 18, no. 10s, pp. 467–481, Jul. 2026.
  - [17]. J. Jagannath, N. Polosky, A. Jagannath, F. Restuccia, and T. Melodia, "Machine Learning for Wireless Communications in the Internet of Things: A Comprehensive Survey," *Ad Hoc Networks*, vol. 93, 101913, 2019, doi: 10.1016/j.adhoc.2019.101913.
  - [18]. C. Koneni and S. Lokula, "Evidence-Quality Telemetry for Cloud Incident Response: Detecting Gaps, Drift, and Integrity Failures in Large-Scale Observability Pipelines," *2026 14th International Symposium on Digital Forensics and Security (ISDFS)*, Boston, MA, USA, 2026, pp. 1-6, doi: 10.1109/ISDFS69419.2026.11459097.
  - [19]. Z. Li, et al., "A Survey of Network Automation for Industrial Internet-of-Things Toward Industry 5.0," *IEEE Transactions on Industrial Informatics*, vol. 19, no. 2, pp. 2065–2077, 2023, doi: 10.1109/TII.2022.3215231.
  - [20]. "Machine Learning Applied to Anomaly Detection on 5G O-RAN Architecture," *Procedia Computer Science*, vol. 222, pp. 81–93, 2023, doi: 10.1016/j.procs.2023.08.146.
  - [21]. E. Habler, R. Bitton, D. Avraham, D. Mimran, E. Klevansky, O. Brodt, H. Lehmann, Y. Elovici, and A. Shabtai, "Adversarial Machine Learning Threat Analysis and Remediation in Open Radio Access Network (O-RAN)," *Journal of Network and Computer Applications*, vol. 236, 104090, 2025, doi: 10.1016/j.jnca.2024.104090.

- [22]. "An Open Testbed for O-RAN Experimentation with AI-Enabled Control and Monitoring," *Internet of Things*, vol. 33, 101729, 2025, doi: 10.1016/j.iot.2025.101729.
- [23]. Vishram Singh, 2026. "Enterprise Resource Planning for Financial Applications: Engineering, Architecture, and Design Principles," *International Journal of Scientific Research in Science and Technology, Technoscience Academy*, vol. 13(2), pages 653-661, April.
- [24]. W. Wu, X. Huang, and T. H. Luan, "AI-Native Network Digital Twin for Intelligent Network Management in 6G," 2024.
- [25]. X. Wang, L. Lu, Q. Li, Q. Sun, N. Shi, Z. Chen, and T. Sun, "A Task-Driven Design Approach for 6G AI-Native Architecture," *Engineering*, vol. 56, pp. 87–103, 2026, doi: 10.1016/j.eng.2025.09.005.
- [26]. "Enabling Mobile AI Agent in 6G Era: Architecture and Key Technologies," *IEEE Network*, vol. 38, no. 5, pp. 66–75, 2024, doi: 10.1109/MNET.2024.3422309.
- [27]. C. Chatzieftheriou and E. Liotou, "A Survey on AI for 6G: Challenges and Opportunities," *IEEE Open Journal of the Communications Society*, vol. 7, pp. 3189–3221, 2026, doi: 10.1109/OJCOMS.2026.3677293.
- [28]. M. Ferrag, et al., "6G Needs Agents: Toward Agentic AI-Native Networks for Autonomous Intelligence," *IEEE Open Journal of the Communications Society*, vol. 7, pp. 7254–7282, 2026, doi: 10.1109/OJCOMS.2026.3707904.
- [29]. P. Kamarajapandian, S. Nithya, A. Syed Haroon, V. Bharathi, A. Kaliappan, and L. Meenachi, "Intent-Based Networking Using LLM-Driven Agentic Reinforcement Learning for Zero-Touch 5G/6G and IoT Systems," *Computer Communications*, vol. 257, 108602, 2026, doi: 10.1016/j.comcom.2026.108602.
- [30]. Koneni, C., S. Lokula, and L. Panjiyar. "A Cloud-Native Approach to Statistical Anomaly Detection and Automated Data Quality Validation." *International Journal of Data Science and IoT Management System 4.4* (2025): 533-543
- [31]. V. K. Bharadwaj Parasaram, V. Teja Bathini and S. K. Nalluri, "An Intelligent Digital Framework for Clinical Trial Monitoring Using AI and Cloud-Based Software Platforms," *2026 2nd International Conference on Artificial Intelligence Systems (AIS)*, San Antonio, TX, USA, 2026, pp. 524-530, doi: 10.1109/AIS69919.2026.11621841.