International Journal of Science and Research (IJSR)
ISSN: 2319-7064
Impact Factor 2025: 7.089

The Role of AI in Network Optimization and Self-Healing Networks

A Technical Review of Predictive, Intent-Driven, and Closed-Loop Network Operations

Vigneshbabu Jeyaseelan[1], Hakkim Rajakani[2]

[1]Cloud, Automation & Cybersecurity Architect, Lead Cloud Engineer (Telecom Private Cloud & AI Infrastructure)
Dallas, Texas, United States

[2]Cloud (Data) Engineer, DevOps & Cloud Automation

Abstract: ***State-of-the-art communication networks are becoming increasingly distributed systems spanning the radio access network, transport network, data center, public cloud, edge, and billions of endpoints. The sheer scale of these networks and the rate at which they are evolving makes manual operations and static, threshold-based management a non-starter. AI and ML offer a way to take vast quantities of telemetry data and turn it into predictions, diagnoses, optimization decisions, and automated remediation. The focus of this article is on the application of AI to network optimization and the development of self-healing networks. We look at the application of AI to traffic engineering, capacity forecasting, anomaly detection, root cause analysis, and closed-loop automation. We relate these capabilities to a number of existing and emerging industry frameworks (3GPP SON, ETSI ZSM, ITU autonomous network architecture, O-RAN AI/ML workflows [1]-[5]). We also propose a practical architecture in which telemetry, network topology, policy guardrails, orchestration, and post-change validation form a continuous feedback loop. The key point is that AI will only create operational value if it is integrated with automation and policy rather than being an isolated analytics layer. When properly governed, AI has the potential to take network operations from a reactive model of incident management to a predictive, adaptive, and increasingly autonomous service assurance model.***

Keywords: Artificial intelligence, machine learning, network optimization, self-healing networks, AIOps, SON, intent-based networking, closed-loop automation, autonomous networks, 5G, O-RAN

1. Introduction

Network engineering has always been predicated on deterministic protocols, capacity models, operational runbooks, and human expertise. That model is still fundamental, just under increasing pressure from the sheer number of variables that modern operators have to manage at any one time. A single digital service can move across a campus or access network, a software-defined WAN, a carrier backbone, cloud load balancers, virtualized network functions, container clusters, and edge. Each of those layers produces metrics, logs, events, config states and dependencies. A failure is therefore rarely a single point failure. A degraded user experience may be the manifestation of congestion, route instability, radio interference, resource overuse, a software bug, a security control, or a dependency somewhere down the chain in another domain.

The networking industry has been moving towards automation for over 10 years. SDN split the control plane from the forwarding plane, and NFV increased the programmability of services. Google's B4 deployment showed the operational benefits of centrally controlled traffic engineering in a large scale production wide area network [6]. In the mobile space, 3GPP formalized the SON concepts and then went on to define self-healing concepts and requirements, providing a very solid basis for automated fault detection and recovery [1], [2]. For 5G 3GPP is continuing the SON management and orchestration work, highlighting the requirement to manage the optimization functions across increasingly complex radio environments [3].

AI takes this evolution from programmable networks to adaptive networks. Instead of just running pre-defined logic, an AI system can estimate future demand, learn what normal operation looks like, rank the likely root causes, and then recommend or select the appropriate corrective action. The ITU autonomous network architecture is the formalization of this broader movement towards networks that can change dynamically with minimal human intervention [4]. ETSI ZSM is similarly focused on end-to-end service management via closed-loop and zero-touch automation, with recent work extending the concept to AI-driven agents and autonomy [5]. The aim of this piece is to describe how AI can be used in network optimization and self-healing without presupposing that autonomy equates to the complete removal of engineering control. A robust self-healing system is not simply an ML model plugged into a config API. It is a governed control loop where data quality, topology context, confidence, policy, change safety, rollback and validation are just as important as model correctness. The rest of the article looks at the drivers for AI adoption, the principal learning techniques, the various optimization use cases, the self-healing architecture, the implementation patterns, the benefits, the risks and the future direction.

2. Why Modern Networks Require AI-Assisted Operations

Three things make modern networks very difficult to run using conventional methods: scale, dynamism, and cross-domain dependency. Scale is in the sheer number of devices, interfaces, sessions, virtual workloads, and telemetry points. Dynamism is due to elastic applications, mobility, cloud autoscaling, constant software releases, and rapidly changing traffic. Cross-domain dependency is that service health is a function of the health of the network, compute, storage, application, identity and security layers.

Threshold alarms are useful for known conditions but are pretty poor at picking up on subtle combinations of signals. A link can be well below a utilization threshold and yet be experiencing microbursts, packet loss, and queuing, which will have a negative impact on the application. Likewise, a single upstream failure can generate hundreds of downstream alarms. AI-based correlation can help to mitigate this event storm by learning the temporal and network relationships. Rather than presenting the operator with every single symptom, they can be grouped, the most likely root cause identified, the customer impact estimated, and the incident ranked.

AI is also of value because the optimization objectives are in conflict with each other. The shortest network path may not be the most resilient; the most resilient path may use up more capacity or energy; the cheapest path to place may breach a latency or regulatory requirement. Autonomous network guidance from the ITU identifies traffic steering, energy saving, mobility optimization, anomaly detection, capacity planning and spectrum management as examples of AI-enabled use cases [7]. These are all multi-dimensional problems which change over time and are therefore well suited to the application of predictive and optimization techniques in the presence of policy constraints.

Another driver is the huge mismatch between machine timescales and human timescales. Radio conditions, congestion, cloud workload placement, etc can all change in seconds or indeed milliseconds. Humans are still ultimately responsible for the architecture, the policy, the exception handling, and the governance, but a lot of the operational decisions have to be made far faster than a human can actually look at them. The right design principle is therefore human-governed autonomy. The engineers define the objectives, the constraints, the approval boundaries, and the rollback conditions, then the automation operates within those boundaries making the repetitive decisions.

3. AI and ML Techniques for Network Optimization

Different network problems call for different AI techniques. Supervised learning works well when you have historical examples that can be labeled (e.g. 'failure in 24 hours', 'congested', 'malicious', 'healthy'). You can use classification models to determine what type of traffic you're looking at or what type of fault you have, and regression models to estimate things like latency, throughput, or future utilization. The main problem with this approach is the cost and quality of the labels. Incident tickets are often incomplete, and the topology at the time of the incident may have changed. Rare events in particular will not have enough examples to train on.

Unsupervised and semi-supervised learning are useful where we don't have labels. Clustering can be used to find groups of devices or flows that behave in a similar manner, whilst anomaly detection can be used to find things that are different from the learned baseline. This is of course particularly relevant when looking for previously unseen operational or security conditions. The caveat is that an anomaly is not necessarily a fault. A successful product launch, a maintenance event or a traffic migration could all be quite anomalous. You therefore need the context of change records, topology, business calendars and service ownership to filter out the false positives.

Time series forecasting is the prediction of future traffic, capacity and performance. The models used can range from statistical models through to recurrent neural networks and transformer based architectures. The forecasts can be used for proactive capacity planning and can also be used as part of a real-time control loop. For instance, the system could forecast that a set of links will be approaching a congestion threshold in 20 minutes and then move some of the flows in advance to prevent users from being impacted.

RL is very appealing for routing and resource scheduling as an agent can learn policies by observing the state, taking an action, and receiving a reward. The reward function could be a combination of throughput, latency, packet loss, energy consumption, and policy compliance. The risk is that a bad reward function will end up driving the wrong behavior. Production RL will therefore benefit from simulation, a constrained action space, digital twins and staged deployment. Deep learning and graph neural networks give us the ability to model the complex positional and network relationships. Generative AI and LLMs are increasingly being looked at as operator copilots and planning agents. ITU's work on generative AI in telecom specifically identifies network modeling, performance analysis, and automated decision making as key future applications [8].

Table 1: AI techniques and their roles in network operations

Technique	Typical Network Inputs	Representative Uses	Operational Considerations
Supervised learning	Labeled alarms, tickets, KPIs, flow records	Fault classification, SLA-risk prediction, traffic classification	Requires reliable labels; retraining needed after topology or software changes
Unsupervised learning	Telemetry, logs, flow behavior	Anomaly detection, clustering, unknown-pattern discovery	Needs contextual validation to avoid false positives
Time-series forecasting	Utilization, latency, demand, radio KPIs	Capacity prediction, congestion avoidance, maintenance planning	Seasonality and sudden regime changes can reduce accuracy
Reinforcement learning	Network state plus action/reward feedback	Routing, scheduling, resource allocation, energy optimization	Must constrain actions and validate safely before production
Graph/deep learning	Topology graphs, multi-dimensional telemetry	Root-cause localization, dependency modelling, complex prediction	Compute intensive; explainability and data alignment are important
Generative/agentic AI	Runbooks, topology, logs, APIs, policies	Operator assistance, plan generation, multi-step remediation	Requires tool permissions, grounding, auditability, and deterministic guardrails

4. AI-Driven Network Optimization

4.1 Traffic Engineering and Dynamic Routing

Traffic engineering is about arranging the flows so that you're making efficient use of network capacity whilst still achieving the required application objectives. The traditional routing protocols we've all been used to work by converging on a solution based on configured metrics and distributed state. SDN introduced the concept of central control and a whole range of new optimization opportunities. Google's B4 work is a very well- known example of large- scale software defined WAN traffic engineering [6]. AI can take this a stage further by allowing you to forecast demand rather than just react to congestion once it's happened. You can then use those forecasts in conjunction with path computation to move elastic traffic, protect your latency sensitive flows, and maintain headroom for failures.

In service provider networks, optimization can be around customer class, latency targets, failure domains, peering costs and maintenance schedules. In the data center context you can look at east-west flows and workload placement. The point is you're not trying to replace the routing protocols, you're trying to provide a higher level of decision making that can be used to select policy, adjust weights, or request paths via the programmable controllers.

4.2 Capacity, Load, and Spectrum Optimization

AI-based capacity management links forecasting to resource allocation. You're no longer just planning from monthly peaks. You can forecast demand by region, by service, by time of day, and by event. In cloud and edge environments, those forecasts can be used to trigger a scale-out prior to demand actually arriving. In a radio network, you can use AI to optimize parameters such as mobility, scheduling, and resource scheduling. O-RAN's architecture explicitly defines the AI/ML workflow responsibilities around the Non-Real Time RAN Intelligent Controller and the policy interaction with the Near-Real Time RIC enabling the learning and optimization functions to influence RAN behavior [9]. O-RAN Release 5 further improved the AI/ML workflow services across the Non-RT and Near-RT RIC environments [10].

The same applies to enterprise Wi-Fi and private 5G. The model can detect an overloaded AP, interference, or poor roaming and will suggest changes to channel, power and/or client steering. These recommendations would of course need to be checked against the local constraints as an optimization which improves aggregate throughput may in fact be detrimental to a key application or a particular site.

4.3 Energy Optimization

Energy has become a first class network optimization objective. The network infrastructure is always on, despite the very significant variation in demand. AI can be used to forecast traffic and put network elements into lower power states where possible and appropriate in terms of redundancy and service level objectives. Ericsson describes AI in the RAN as a means of improving performance, automation and energy efficiency by distributing intelligence across the radio, RAN compute, Cloud RAN and management layers [11]. The optimization problem has to take into account wake-up time, coverage, resilience, hardware wear, and user experience. You wouldn't want to go too far down the energy savings route at the expense of fault tolerance.

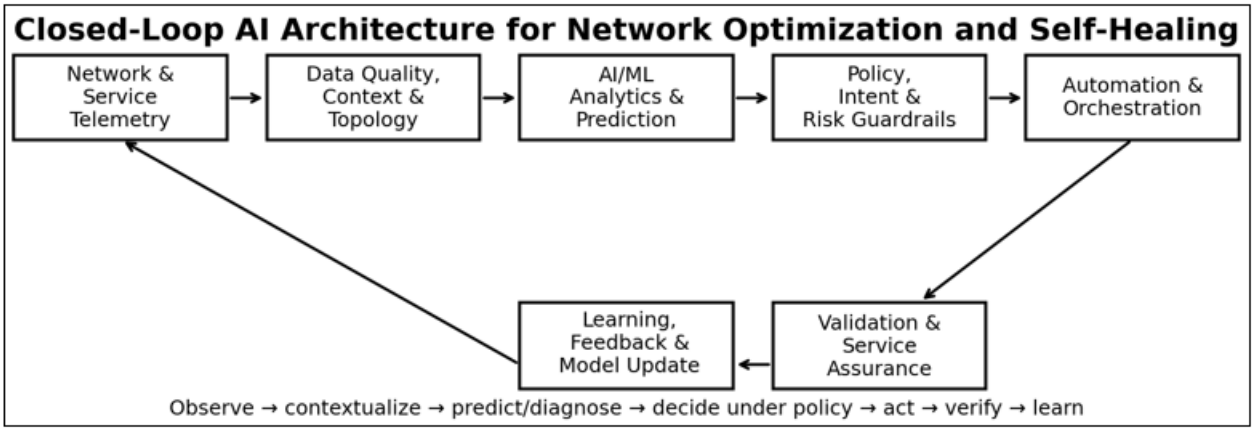


Figure 1: Reference closed-loop architecture for AI-assisted optimization and self-healing

5. Self-Healing Networks and Closed-Loop Automation

A self-healing network is able to detect a deviation, determine the appropriate corrective action, carry out that action, and verify that recovery has occurred with minimal human intervention. 3GPP's self-healing specification makes self-healing a formal SON management function [2]. The concept is now being extended well beyond the radio access network to cover end-to-end services across transport, virtualized infrastructure, cloud, and security.

Stage 1 is observation. Telemetry can be anything from streaming counters and logs, traces, routing state, flow data, synthetic tests, radio measurements, environmental sensors to configuration snapshots. Stage 2 is context. Telemetry is vastly more useful when it's contextualized with topology, dependency graphs, maintenance windows, software versions, ownership, and service criticality. Stage 3 is detection and diagnosis. The models detect anomalies, correlate symptoms, and generate one or more root cause hypotheses with associated levels of certainty.

Stage 4 is decision. A policy engine decides whether the proposed action is permitted. A low risk action (e.g., restarting a stateless service instance) may be fully automated, whereas a routing change which would impact a large number of customers may require approval. Stage 5 is execution via SDN controllers, orchestration platforms, APIs, infrastructure as code, or device automation. Finally, the system checks that the service indicators have improved and that no new violation has been introduced. If validation fails, then the workflow can roll back or escalate.

5.1 Root-Cause Analysis and Event Correlation

Root cause analysis is hard because network alarms are so highly correlated. A single fiber cut can result in interface alarms, routing changes, tunnel failures, application timeouts, and customer tickets. Graph based correlation uses the topology and service dependencies to work out the propagation paths. ML can add a probabilistic ranking based on historical incidents. A useful system would provide the evidence (the sequence of events, the affected dependencies, the similar prior incidents) rather than just return a root cause label. Explainability is critical when the recommendation is going to result in an automated change.

5.2 Remediation Patterns

Self-healing generally works in a very limited number of patterns: reroute around a failed/degraded path, restart or replace a failed software function, scale up, roll back a config, isolate a bad node, tweak a radio or QoS parameter, or open a human work item if you need to do physical intervention. The safest way to deploy is to start with recommendation mode, then move to approval gated execution, and only then go to full automation for actions which have very well understood failure modes. Each playbook must define the prerequisites, the blast radius, the success criteria, the timeout behavior, and the rollback.

5.3 Intent-Based and Zero-Touch Operations

Intent-based networking shifts the unit of management from device commands to the desired outcome. An intent might be that a payment service must meet a certain latency objective, use approved security domains and have two independent paths. The controller takes that intent and translates it into configuration, then continually compares the observed state against the desired state. ETSI ZSM applies the same closed-loop principles to end to end service management and is very much on the path to AI-driven agentic architectures, intent-based automation, and cross-domain orchestration [5]. The ITU autonomous network framework provides a complementary architectural view of increasing levels of network autonomy [4].

Table 2: Closed-loop self-healing stages and operational controls

Stage	Primary Function	Example AI/Automation Role	Required Guardrail
Observe	Collect health and service signals	Detect shifts in latency, loss, utilization, radio quality	Timestamp quality, completeness, source identity
Diagnose	Correlate symptoms and infer cause	Rank probable failing link/device/service dependency	Topology awareness, confidence threshold, explainability
Decide	Select corrective action	Choose reroute, scale, restart, rollback, or escalation	Policy, maintenance state, risk and blast-radius limits
Act	Execute change	Controller/API/playbook performs remediation	Least privilege, idempotency, transaction logging
Verify	Measure post-change state	Confirm SLA/KPI recovery and absence of regression	Explicit success criteria and rollback timeout
Learn	Improve future decisions	Update features, incident similarity, model performance	Versioning, drift detection, approval for model promotion

6. Reference Architecture for Production Deployment

A production architecture should separate analytics from control in such a way that the model isn't going to be able to just go and make arbitrary network changes. The telemetry layer is where the high frequency data is collected and then pushed through streaming or event- based pipelines. There's a data platform which stores both the real-time and the historical data and which then enriches that with inventory, topology, configuration, incident, and service data. There are feature pipelines which create the consistent input that you need for training and inference. The model services then output forecasts, anomaly scores, classifications, and action recommendations.

Above the model layer, there's a decision service which combines the model output with a deterministic policy. That's where the confidence thresholds, the maintenance windows, the service criticality, the change freezes, the regulatory limits and the approval requirements all get evaluated. The orchestration layer then carries out the approved actions via the controllers and automation systems. There's a validation service which measures the post-change behavior and closes the loop. Every decision and every action must be logged with the model version, the input evidence, the policy result, the executor identity, and the outcome.

Multi-vendor interoperability is a must. The various standards bodies are all working on it in their own way: 3GPP has defined the management and SON functions for mobile networks [1]-[3]; O-RAN has defined the RIC and AI/ML workflow interfaces for open RAN [9], [10]; ETSI ZSM has defined an end-to-end zero-touch management framework [5]; and ITU has defined the architectural concepts and use cases for autonomous networks [4], [12]. The approach should be to use these schemes to reduce the amount of proprietary coupling while still allowing the vendor specific optimisations to be retained where they are of value.

Model lifecycle management has to be part of network lifecycle management. Models need to be versioned, tested against representative topologies and failure cases, monitored for drift, and promoted through the various environments. O-RAN's work on AI/ML testing identifies compatibility, scalability, uncertainty, safety, privacy, security, and trust as the key issues when AI is being brought into the network [13]. Those are all principles that apply beyond the RAN.

7. Operational Benefits and Business Impact

The most obvious benefit of AI-assisted operations is faster detection and diagnosis. Event correlation can reduce the number of alarms that get to the operator, and predictive models can flag degradation before a hard failure occurs. When diagnosis is tied to definitive remediation, you can see a massive reduction in both MTTD and MTTR. The operating model goes from being ticket driven to being service driven.

AI can also improve utilization. Predictive traffic engineering enables operators to make better use of their available capacity whilst maintaining resilience margins. Capacity forecasts will allow better timing of upgrades and thus reduce both over-provisioning and the need for emergency expansion. Energy aware optimisation can also lead to reduced OPEX and reduced environmental impact. Ericsson's current AI networking material describes dynamic resource allocation and traffic prediction as the two key mechanisms for improving capacity utilization, reliability and the end user experience [14].

Another benefit is consistency. An experienced engineer will have an implicit understanding of the various failure modes that they've seen over the years. That knowledge can be captured in features, policies, and remediation workflows so it's available 24/7 to all teams. It doesn't remove the expertise, it just enables the experts to focus on architecture, the odd incident, policy, and the continual improvement of the system. AI can also help with knowledge retrieval by linking incidents to runbooks, previous changes, topology, and known defects.

For the business stakeholder, the value is in improved service availability, fewer incidents that impact the customer, faster time to service delivery, more predictable capacity spend, and less operational effort. That value has to be measured directly. A self-healing initiative shouldn't be judged on model accuracy alone. It should be judged on avoided incidents, false action rate, restoration time, change success, energy saved, SLA improvement, and operator time saved.

8. Challenges, Security, and Governance

AI introduces new failure modes. Bad data quality can lead to incorrect inference, and model drift can slowly erode accuracy as traffic, software, or topology changes. The training data may contain operational biases as well as historical misdiagnoses. An attacker who can manipulate telemetry, model inputs, or automation credentials may be able to influence the decisions. Closed-loop automation therefore extends the security boundary from the network devices out to the data pipelines, model repos, feature stores, orchestration systems, and AI agents.

Explainability is a practical requirement rather than just an academic one. The operator needs to know why an action is being recommended, what the evidence was that led to that recommendation, how confident the system is in that recommendation, and what the alternative hypotheses were. High impact actions will of course need higher standards of evidence. Where generative or agentic AI is being used, tool access must be allow listed and restricted. Generated plans must be translated into a set of validated, deterministic actions rather than allowing the AI to generate the actual commands unconstrained.

Blast radius management is just as important. Autonomy must be progressive. Orgs can define various levels of autonomy (e.g., observe only, recommend, human approved execute, bounded autonomous execution, cross domain autonomous optimization). Promotion to a higher level of autonomy is dependent on demonstrated reliability. Canary execution, simulation, digital twins, maintenance window awareness, and automatic rollback all serve to reduce the risk of a large blast radius.

Governance also requires clear ownership. Network engineering owns the service intent and technical constraints; security owns the access and threat controls; data and AI own the model quality; and operations own the execution and incident processes. The audit records need to be sufficient to allow a reviewer to reconstruct every automated decision. ETSI's current ZSM program has explicit work in the area of closed loop automation security, which again underlines the importance of securing the autonomous control paths [5].

9. Implementation Roadmap and Assessment Criteria

Organizations should treat the introduction of AI-driven network autonomy as a staged engineering exercise rather than a single platform deployment. The first stage is observability readiness. You need to be able to accurately inventory your network, have a good topology, good configuration, good service ownership, and good telemetry. The objective of this stage is to have common identifiers across your monitoring, your ticketing, your orchestration, and your configuration systems so that events from different systems can be joined to the same device, interface, service, or customer dependency. You need to be able to measure your data quality objectives explicitly (telemetry completeness, timestamp accuracy, topology freshness, % of incidents that can be mapped to a service dependency, etc.).

Phase 2 is decision support. The models are operating in shadow/recommendation mode and are being compared against the actual decisions of the operator. This is the right phase to be looking at precision, recall, false positive rate, calibration, and root cause ranking quality. The technical model metrics need to be linked to operational outcomes though. A model that has very good statistical accuracy but generates hundreds of low value alerts is not going to be of any use in terms of actual operations. More useful metrics would be the reduction in the number of alerts, the % of incidents where the top 3 root cause hypotheses are correct, the time saved during triage, and the % of predicted capacity risks that lead to some form of useful preventative action.

Phase 3 is approval gated remediation. The AI comes up with a recommendation, builds the evidence, and then a human has to approve the action before it is executed. That validates the full lifecycle of the workflow (diagnosis, policy assessment, automation, verification, and rollback) without giving the AI any actual autonomy. Actions are going to be risk classified. Low risk, reversible actions can move through the process much more quickly. High risk actions (i.e. high impact routing, firewall, radio, customer facing changes, etc.) will require much more simulation and more rigorous change controls. A remediation is not considered a success simply because the automation has run. Success is when the service indicators in question have returned to an acceptable state, and there has been no significant regression elsewhere.

Phase 4 enables bounded autonomy for proven use cases. The boundaries can be expressed in a number of ways: maximum number of devices impacted, maximum number of configuration objects that can be changed, service criticality, geography, time window or confidence threshold. A system might be permitted to automatically restart one failed stateless network function or automatically reroute a small class of traffic, whilst being prohibited from changing backbone policy, for example. This provides the benefit of automation at speed without creating an unbounded control surface.

Operational scorecards should incorporate reliability, safety, efficiency and business metrics. Reliability metrics would be mean time to detect, mean time to isolate, mean time to restore, incident recurrence, and SLA attainment. Safety metrics would be false remediation rate, rollback rate, change failure rate, policy violations, and the largest blast radius observed. Efficiency metrics would be the number of alarms suppressed via correlation, operator hours saved, capacity utilization, energy consumption and automation completion time. Business metrics could be avoided customer impacting minutes, reduced escalation volume, improved service availability and deferred capex. Model governance would add a further set of metrics including drift, inference latency, confidence calibration, data freshness and performance by network domain.

Ultimately, organizations should have an autonomy review process. New models and new remediation policies should go through lab or digital twin testing, controlled production canaries, and post-deployment review. Incident retrospectives should include a review of whether the AI recommendations were correct, whether the automation behaved as expected, and whether the guardrails were adequate. The end result is a learning system at two levels: the models learn from the data and the operating organization learns which decisions are safe to hand over. This is a credible path from AI-assisted operations to self-healing networks without undermining engineering accountability.

10. Future Directions

The next stage of AI networking is going to be the integration of specialist predictive systems with generative and agentic systems. The specialist models will still be doing the high speed stuff (forecasting, anomaly scoring, radio optimization, etc.). The agentic systems will be the ones that can orchestrate longer workflows; they can go out and gather evidence from a number of different domains, form a hypothesis, query a digital twin, select a remediation plan, request approval where appropriate, execute via the tools, and then summarise the result. ETSI is already looking at agents in autonomous networks, and ITU has work in progress on the protocols and signaling requirements for agentic AI assistance [5], [15].

Digital twins are going to become increasingly important as autonomous systems will need safe environments in which to experiment. A network twin can model the network topology, the demand, the policies, and the failure scenarios, enabling an optimization policy to be tested prior to it being used in production. Federated learning can allow operators to learn across multiple distributed sites without the need to centralize all the raw data. Edge inference will move the decision making much closer to the radio and the industrial endpoint when either latency or data sovereignty makes centralized inference impractical.

AI-native network concepts go further by treating AI as a fundamental element right through the network lifecycle (design, deployment, operation and maintenance) rather than just as an external analytics tool [16]. This could see the model lifecycle, inference placement, data quality, and agent coordination all become first class network functions. Deterministic protocols and formal policy will of course still be required. The likely end state is a hybrid architecture where AI is responsible for dealing with uncertainty and prediction and the traditional control systems are responsible for safety and correctness.

11. Conclusion

AI is becoming a fundamental technology for network optimization and self-healing as it is starting to deal with problems that are simply not amenable to static thresholding and manual workflows anymore. Supervised and unsupervised learning can be used to classify faults and detect anomalies; forecasting can be used to anticipate congestion and capacity demand; reinforcement learning can be used to make optimal sequential resource decisions; graph and deep learning can be used to reason across complex dependencies; and generative or agentic AI can be used to assist with multi-step business workflows.

The best architecture isn't one where the AI is in complete control of the network. It's one where the AI is part of a closed loop that is driven by intent, policy, risk controls, deterministic automation, and post-change validation. That's the architecture that takes you from observability through to recommendation, bounded remediation, and then progressively higher levels of autonomy. The 3GPP, ETSI, ITU and O-RAN standards all demonstrate that this is the emerging structure of next generation network management rather than a set of disparate vendor features [1]-[5].

Organizations that take this approach can improve availability, utilization, energy efficiency and operational consistency by removing the need for repetitive manual work. Success is predicated on good data engineering, good model lifecycle management, security, explainability and very careful control of the automation blast radius. Self-healing is therefore an engineering system rather than just an AI model.

When intelligence, automation and governance are designed together, the network can move from a reactive infrastructure to an interactive one that can anticipate degradation, continuously optimize its resource usage and recover from a wide variety of failures well before the user is even aware of any problem.

References

[1] 3GPP, TS 32.500, "Telecommunication management; Self-Organising Networks (SON); Concepts and requirements", current specification.
[2] 3GPP, TS 32.541, "Telecommunication management; Self-Organising Networks (SON); Self-healing concepts and requirements", current specification.
[3] 3GPP, TS 28.313, "Management and orchestration; Self-Organising Networks (SON) for 5G networks", current specification.
[4] ITU-T, Recommendation Y.3061, "Autonomous networks, Architecture framework," Dec. 2023.
[5] ETSI, "Zero-touch network and Service Management (ZSM)", technical group and standards program, accessed 2026.
[6] S. Jain et al., "B4: Experience with a Globally-Deployed Software Defined WAN", Proc. ACM SIGCOMM, 2013.
[7] ITU-T, Recommendation Y.2361, "Requirements for fostering telecommunication/ICT services universalization using open networks and AI models", April 2025.
[8] ITU-T, TR.GenAI-Telecom, "Potential requirements and methodology for deploying and assessing Generative AI models in telecom networks", agreed Mar. 2025.
[9] O-RAN Alliance, WG2, "Non-Real-time RAN Intelligent Controller and A1" workgroup scope (includes AI/ML workflow and policy- based optimisation).
[10] O-RAN Alliance, "O-RAN Alliance Completed its Specification Release 5 (O-RAN-R005)", 8 June 2026.
[11] Ericsson, "AI RAN - Smarter connectivity", 2026.
[12] ITU-T, Supplement 71 to Y-series Recommendations, "Use cases for autonomous networks", Jul. 2022.
[13] O-RAN Alliance, "Principles and Methodologies for AI/ML Testing in Next Generation Networks", research report.
[14] Ericsson, "Telecom AI - Networks for AI and AI for networks", 2026.
[15] ITU-T, QSTR.ANAI, "Survey and analysis of available protocols for autonomous network with AI assistance", work item, 2026.
[16] ITU-T Focus Group on AI-Native Networks, "Glossary of terms and definitions for Autonomous Networks", 2025.