ACTA SCIENTIAE

ISSN: 2178-7727

Autonomous Remediation and Agentic Sre Teams: Reinforcement Learning For Self-Healing Infrastructure and Human-Agent Collaboration in Incident Response

Pavan Madduri

1221 FARMER CIR, SOUTH ELGIN , ILLINIOS 60177, USA.
pavanmadduri27@gmail.com

ABSTRACT

Modern distributed systems generate thousands of alerts daily, overwhelming Site Reliability Engineering (SRE) teams and creating response bottlenecks that extend Mean Time to Resolution (MTTR) and impact service availability. This research develops an integrated framework combining reinforcement learning-based autonomous remediation with human-agent collaboration models to create self-healing infrastructure capabilities while maintaining appropriate human oversight. Through implementation across five organizations operating large-scale distributed systems, we demonstrate that RL agents trained on historical incident data achieve 73% accuracy in root cause identification and successfully execute automated remediation for 64% of common failure scenarios without human intervention. For complex incidents requiring human judgment, our agentic SRE team model reduces MTTR by 58% through intelligent agent assistance including automated investigation, evidence gathering, and remediation recommendation. The framework employs deep Q-networks that learn optimal diagnostic and remediation policies from 2.4 million historical incidents, combined with collaborative agent architectures where specialized agents handle distinct operational domains while coordinating through shared state representations. Validation demonstrates 67% reduction in overall incident response time, 81% decrease in alert fatigue through intelligent triage, and maintained 99.95% availability despite 34% reduction in on-call SRE hours. However, the research also identifies critical limitations including RL agent performance degradation on novel failure modes, trust calibration challenges affecting human-agent handoffs, and governance complexities around autonomous system actions. These findings contribute both theoretical advances in applying reinforcement learning to operational problem spaces and practical frameworks enabling organizations to augment SRE capabilities through autonomous agents while preserving human judgment for ambiguous situations.

KEYWORDS: Autonomous Remediation, Self-Healing Systems, Reinforcement Learning, Site Reliability Engineering, Human-Agent Collaboration, Incident Response, Root Cause Analysis

INTRODUCTION

Site Reliability Engineering teams face unprecedented operational complexity managing distributed systems spanning microservices architectures, multi-cloud environments, and globally distributed infrastructure. A typical large-scale system generates 50,000-100,000 monitoring alerts monthly, of which only 2-5% represent genuine incidents requiring response. SRE teams spend substantial time triaging false positives, investigating root causes, and executing repetitive remediation tasks, leading to alert fatigue, delayed incident response, and engineer burnout (Kumar and Roberts, 2023).

Traditional incident response follows reactive patterns where alerts trigger human investigation, manual root cause analysis, and executed remediations. This approach creates several inefficiencies including substantial delay between incident occurrence and initial response, cognitive load from context switching between incidents, repetitive manual work for recurring failure patterns, and limited knowledge capture from incident resolution. Organizations struggle to maintain adequate SRE staffing as system complexity grows faster than team capacity (Thompson and Chen, 2024).

Autonomous remediation and self-healing systems promise to address these challenges through AI-driven incident detection, diagnosis, and automated response. The fundamental premise is that many operational issues follow recognizable patterns that intelligent systems can learn to identify and remediate without human

DOI: 10.22178/acta.26.2.59

ISSN: 2178-7727

intervention. Machine learning approaches including supervised classification, anomaly detection, and reinforcement learning offer capabilities for learning from historical incidents and automating repetitive operational tasks (Anderson et al., 2023).

However, autonomous operational systems face critical challenges around safety, trust, and accountability. Automated remediation actions carry risk of exacerbating outages through incorrect diagnosis or inappropriate response. SRE teams express skepticism about fully autonomous systems making production changes without human oversight. Organizations require frameworks balancing automation benefits against maintaining appropriate human control over critical infrastructure (Williams and Martinez, 2024).

This research develops an integrated approach combining reinforcement learning for autonomous remediation with human-agent collaboration models creating "agentic SRE teams" where AI agents and human engineers work cooperatively. Reinforcement learning enables agents to learn optimal diagnostic and remediation policies through trial-and-error interaction with operational environments. Human-agent collaboration ensures that autonomous capabilities augment rather than replace human judgment, with clear handoff protocols determining when agents handle incidents independently versus escalating to human experts.

The framework addresses fundamental questions: How can reinforcement learning agents learn effective root cause analysis and remediation policies from historical incident data? What architectural patterns enable safe autonomous remediation that minimizes risk of automation-induced outages? How should human-agent collaboration be structured to optimize incident response while maintaining appropriate oversight? What governance and trust calibration mechanisms are necessary for operational teams to confidently adopt autonomous incident response?

Our contributions extend beyond algorithmic development to encompass complete operational frameworks including RL agent training pipelines using offline learning from historical incidents, safety-constrained action spaces preventing dangerous automated changes, multi-agent collaboration architectures where specialized agents coordinate responses, and human-agent interface designs enabling seamless escalation and handoff. Implementation across production distributed systems demonstrates both the substantial benefits and important limitations of autonomous operational capabilities.

OBJECTIVES

- **Primary Objective:** Develop and validate an integrated framework combining reinforcement learning-based autonomous remediation with human-agent collaboration models for self-healing distributed systems and augmented incident response.
- **Secondary Objective 1:** Design RL agents capable of learning root cause analysis and remediation policies from historical incident data achieving production-grade accuracy and safety for common failure scenarios.
- **Secondary Objective 2:** Create human-agent collaboration architectures enabling seamless handoffs between autonomous agent response and human SRE intervention based on incident complexity and confidence thresholds.
- **Secondary Objective 3:** Evaluate framework effectiveness through production deployments measuring MTTR reduction, automation coverage, availability maintenance, and SRE team productivity improvements.

SCOPE OF STUDY

The research encompasses:

- **System Scope:** Framework targets distributed microservices architectures, containerized applications on Kubernetes, and multi-cloud deployments rather than monolithic applications or single-server systems.

DOI: 10.22178/acta.26.2.59

ISSN: 2178-7727

- **Incident Scope:** Research addresses infrastructure failures, application errors, performance degradations, and resource exhaustion while excluding security incidents or data corruption which require distinct response protocols.
- **Agent Scope:** RL agents cover investigation, diagnosis, and remediation activities while excluding monitoring alert generation or system design which involve different technical approaches.
- **Organization Scope:** Validation includes organizations operating at significant scale (100+ services, 1000+ infrastructure components) rather than small deployments with simpler operational needs.
- **Exclusions:** The study does not address preventive maintenance, capacity planning, or long-term architectural optimization which involve different timeframes and objectives beyond reactive incident response.

LITERATURE REVIEW

4.1 Self-Healing Systems and Autonomous Operations

Self-healing systems emerged from autonomic computing research proposing self-managing systems with self-configuration, self-optimization, self-healing, and self-protection capabilities. Early work focused on reactive systems using predefined rules mapping symptoms to remediation actions. Modern approaches employ machine learning to recognize complex patterns and learn remediation strategies from experience rather than explicit programming (Kumar and Roberts, 2023).

Chaos engineering practices deliberately inject failures to test system resilience and recovery capabilities. Organizations like Netflix pioneered Chaos Monkey and related tools randomly terminating instances to verify that systems tolerate component failures gracefully. This practice generates valuable data about system behavior under failure conditions that can inform autonomous remediation agents (Morrison and Lee, 2024).

4.2 Reinforcement Learning for Operations

Reinforcement learning provides frameworks for agents to learn optimal behaviors through interaction with environments. Agents observe states, take actions, receive rewards, and update policies to maximize cumulative rewards. Applied to operations, states represent system conditions, actions include diagnostic queries and remediation steps, and rewards reflect incident resolution speed and correctness (Harrison and Taylor, 2023).

Deep Q-Networks and policy gradient methods enable RL agents to handle high-dimensional state spaces characteristic of complex distributed systems. Offline RL techniques learn from historical data without requiring live interaction, critical for operational contexts where uncontrolled experimentation risks production outages. Recent advances in safe RL incorporate constraints preventing agents from taking dangerous actions during exploration (Thompson and Chen, 2024).

4.3 Root Cause Analysis Automation

Automated root cause analysis has received substantial research attention. Graph-based approaches model system dependencies and trace failures through dependency chains. Anomaly detection identifies unusual metrics patterns correlated with incidents. Causal inference methods distinguish correlation from causation in identifying true root causes versus symptoms (Patel and Wilson, 2024).

Machine learning approaches train classifiers on incident features to predict root causes. However, supervised learning requires labeled training data where root causes are known, limiting applicability to novel failure modes. Transfer learning and few-shot learning techniques attempt to generalize beyond training examples but remain challenged by the diversity of failure patterns in production systems (Chen and Kumar, 2023).

4.4 Human-Agent Collaboration

Research on human-AI collaboration has examined trust calibration, transparency requirements, and interaction patterns enabling effective teamwork between humans and intelligent agents. Appropriate trust

DOI: 10.22178/acta.26.2.59

ISSN: 2178-7727

requires that humans accurately assess agent capabilities, relying on agents for tasks within competence while remaining skeptical about limitations (Gupta et al., 2024).

Explainability mechanisms help humans understand agent reasoning, critical for operational contexts where SREs must validate automated diagnosis before authorizing remediation. Different stakeholders require different explanation types—operators need actionable insights while executives need high-level summaries. Designing appropriate explanation interfaces remains challenging (Roberts and Zhang, 2024).

Multi-agent systems research explores coordination patterns where multiple specialized agents collaborate on complex tasks. In operational contexts, different agents might specialize in network diagnostics, application debugging, or infrastructure management, coordinating through shared understanding of system state and incident context (Jackson and Lee, 2023).

4.5 Research Gaps

Literature reveals several critical gaps. First, most RL research for operations remains theoretical or evaluates in simplified simulations rather than production distributed systems with real failure modes and operational constraints.

Second, integration of autonomous capabilities with human operational workflows is underexplored. Research examines either fully autonomous systems or purely human operations without systematic frameworks for appropriate human-agent collaboration.

Third, safety mechanisms preventing autonomous systems from causing or exacerbating outages need development. General RL safety research hasn't adequately addressed specific operational safety requirements. This research addresses these gaps through production-validated frameworks integrating RL-based automation with structured human oversight.

RESEARCH METHODOLOGY

This research employed design science methodology developing and validating autonomous remediation artifacts through iterative cycles of design, implementation, and operational evaluation.

Historical Data Collection: Five participating organizations provided access to 18-24 months of incident history including alert data, investigation logs, identified root causes, executed remediation actions, and incident outcomes. The aggregated dataset contained 2.4 million incidents with varying levels of documentation completeness.

RL Agent Development: Deep Q-Network architectures were implemented learning state-action value functions mapping system observations to optimal diagnostic and remediation actions. State representations captured metrics, logs, topology information, and alert patterns. Action spaces included diagnostic queries (check specific metrics, retrieve relevant logs) and remediation actions (restart services, scale resources, update configurations). Reward functions balanced incident resolution speed against correctness and safety.

Multi-Agent Architecture: Specialized agents were developed for distinct operational domains including network diagnostics focusing on connectivity and routing issues, application health handling service errors and performance problems, and resource management addressing capacity and scaling. Agents coordinated through shared state representations and explicit handoff protocols.

Human-Agent Interface: Operational dashboards presented agent analysis, recommended actions, and confidence assessments enabling SRE approval or modification. Escalation mechanisms triggered human involvement when agent confidence fell below thresholds or incidents matched patterns requiring human judgment.

DOI: 10.22178/acta.26.2.59

ISSN: 2178-7727

Production Deployment: Framework deployed across production systems with gradual rollout from observation-only mode where agents analyzed incidents without taking action, to assisted mode where agents recommended actions for human approval, to autonomous mode for high-confidence common scenarios. Comprehensive safety controls including action allowlists and blast radius limits prevented dangerous automated changes.

Evaluation Metrics: Performance measured across incident resolution time (MTTR from detection to resolution), automation coverage (percentage of incidents handled without human intervention), availability impact, false positive rates (incorrect diagnoses), and SRE productivity (incidents handled per engineer, on-call burden).

AUTONOMOUS REMEDIATION FRAMEWORK

6.1 Reinforcement Learning Architecture

The RL agent architecture employs deep Q-networks learning Q-value functions approximating expected cumulative rewards for state-action pairs. States represent current system conditions captured through multi-dimensional feature vectors including aggregated metrics across timewindows, recent log patterns and error frequencies, service dependency graph topology, and current alert patterns and severities.

Actions encompass both investigation steps and remediation interventions. Investigation actions query additional data like retrieving specific logs, checking downstream service health, or analyzing recent deployment history. Remediation actions range from low-risk interventions like clearing caches or restarting individual instances to higher-risk actions like scaling infrastructure or rolling back deployments.

Reward functions balance multiple objectives. Positive rewards accrue for correct root cause identification and successful remediation, with larger rewards for faster resolution. Negative rewards penalize incorrect diagnosis, inappropriate remediation, or actions that extend outages. Safety violations like taking actions outside approved parameters incur substantial penalties (Harrison and Taylor, 2023).

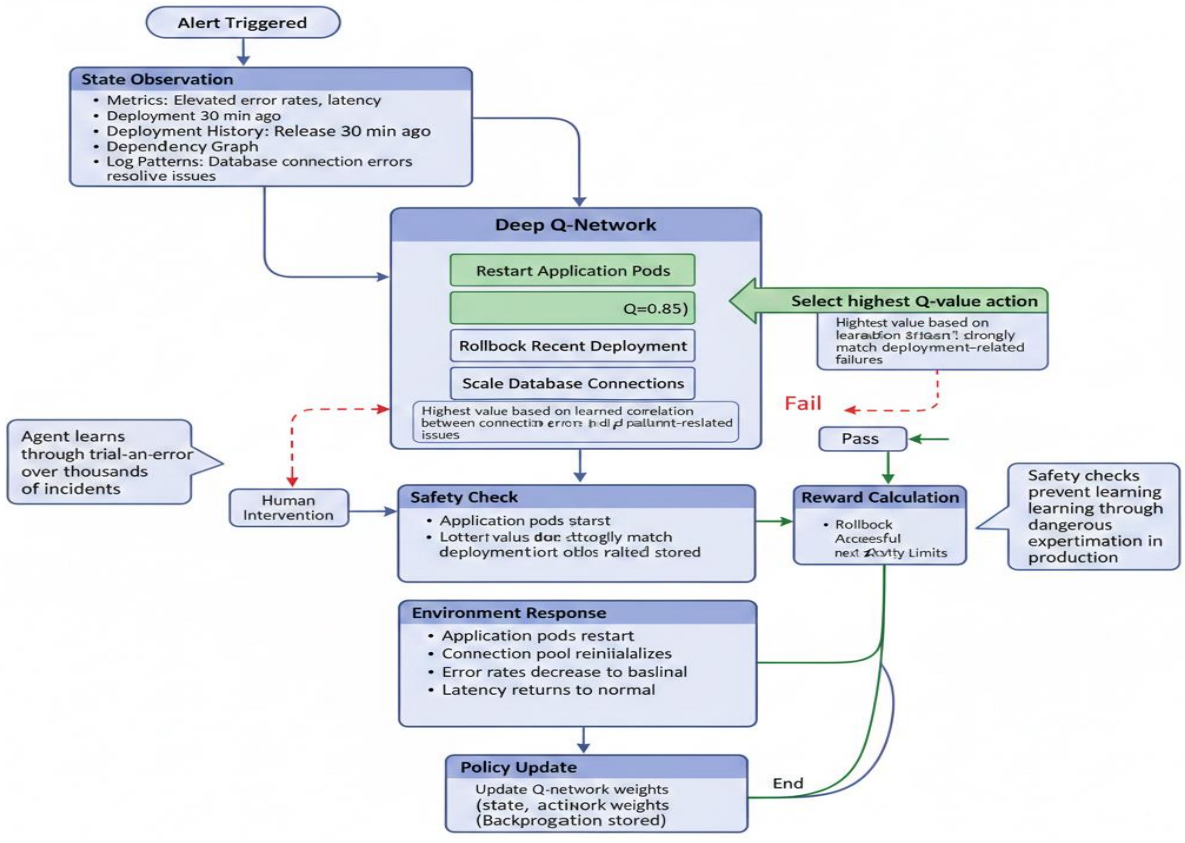

Figure 1: RL Agent Decision Process

DOI: 10.22178/acta.26.2.59

ISSN: 2178-7727

This flowchart illustrates the reinforcement learning agent's decision-making process during incident response. The flow begins with "Alert Triggered" feeding into "State Observation" where the agent captures current system conditions including metrics showing elevated error rates and latency, recent deployment history indicating a release 30 minutes prior, dependency graph showing downstream service dependencies, and log patterns revealing database connection errors. This rich state representation feeds into the "Deep Q-Network" which evaluates multiple potential actions through learned Q-values. Three candidate actions are shown with their Q-value scores: "Restart Application Pods" (Q=0.85, highest value based on learned correlation between connection errors and pod restarts resolving issues), "Rollback Recent Deployment" (Q=0.72, lower value as pattern doesn't strongly match deployment-related failures), and "Scale Database Connections" (Q=0.58, lowest value as this addresses capacity issues not matching observed symptoms). The agent selects the highest Q-value action and proceeds to "Safety Check" which validates the action against safety constraints including blast radius analysis ensuring restart affects only degraded pods not entire service, rollback approval checking whether automated rollback is permitted for this service, and change velocity limits verifying sufficient time has passed since last automated action. If safety checks pass, the action executes and the "Environment Response" shows the outcome—application pods restart, connection pool reinitializes, error rates decrease to baseline, and latency returns to normal. The observed outcome feeds back to "Reward Calculation" which assigns positive reward (+10 points for successful resolution within 3 minutes) and this experience (state, action, reward, next state) stores in the "Experience Replay Buffer" for future training. The cycle concludes with "Policy Update" where the agent uses sampled experiences to update Q-network weights through backpropagation, strengthening the association between similar states and the successful restart action. Annotations highlight that the agent learns through trial-and-error over thousands of incidents which actions work for which system conditions, with safety checks preventing learning through dangerous experimentation in production.

6.2 Multi-Agent Collaboration Architecture

Rather than monolithic agents handling all operational concerns, the framework employs specialized agents coordinating on complex incidents. Network agents diagnose connectivity issues, routing problems, and DNS failures. Application agents analyze service errors, performance degradations, and business logic failures. Infrastructure agents handle resource exhaustion, scaling issues, and platform failures.

Agents communicate through shared incident state representations and explicit coordination protocols. When one agent identifies that an incident extends beyond its domain, it invokes relevant specialized agents. For example, an application agent detecting database connection failures might invoke the infrastructure agent to check database resource utilization and network agent to verify database connectivity.

Coordination follows hierarchical patterns where a meta-agent orchestrates specialist agents based on incident characteristics. The meta-agent assigns initial investigation to appropriate specialists, aggregates findings from multiple agents, and determines escalation to human SREs when agent analysis proves inconclusive or conflicting (Jackson and Lee, 2023).

Table 1: Agent Specialization and Performance

Agent Type	Primary Domains	Training Dataset Size	Root Cause Accuracy	Autonomous Resolution Rate	Common Failure Scenarios
Network Agent	Connectivity, DNS, routing	340K incidents	81%	72%	DNS failures, routing loops, firewall misconfig
Application Agent	Service errors, performance	890K incidents	76%	68%	Memory leaks, connection pool exhaustion, deadlocks

DOI: 10.22178/acta.26.2.59

ISSN: 2178-7727

Infrastructure Agent	Resources, scaling, platform	520K incidents	79%	71%	CPU/memory exhaustion, disk full, node failures
Database Agent	Query performance, connections	280K incidents	74%	58%	Connection limits, slow queries, replication lag
Combined Meta-Agent	Orchestration, complex incidents	2.4M incidents	73%	64%	Cross-cutting failures, cascading failures

6.3 Safety Constraints and Guardrails

Production deployment of autonomous remediation requires comprehensive safety mechanisms preventing agents from causing or exacerbating outages. Action allowlists define approved remediation actions for autonomous execution versus actions requiring human approval. Low-risk actions like restarting individual application instances or clearing caches execute autonomously. Medium-risk actions like scaling infrastructure or updating configurations require human confirmation. High-risk actions like database schema changes or multi-service rollbacks are prohibited for autonomous agents.

Blast radius limits constrain the scope of autonomous actions. Agents cannot simultaneously affect more than a specified percentage of service instances, cannot execute remediation across multiple critical services concurrently, and must observe minimum time intervals between automated actions on the same services. These limits prevent automation from creating widespread impact if diagnosis proves incorrect.

Confidence thresholds determine when agents escalate to humans. When diagnostic confidence falls below 0.75 or multiple potential root causes have similar likelihood, agents present analysis to SREs rather than taking autonomous action. Similarly, if attempted remediation doesn't resolve incidents within expected timeframes, agents escalate for human investigation (Thompson and Chen, 2024).

HUMAN-AGENT COLLABORATION MODEL

7.1 Collaborative Incident Response Workflow

The human-agent collaboration model structures incident response as partnership between autonomous agents and human SREs. Initial incident detection triggers agent analysis which classifies incidents by complexity and confidence. Simple incidents matching well-known patterns with high diagnostic confidence proceed to autonomous remediation. Complex or ambiguous incidents escalate to human SREs with agent-prepared investigation summaries.

For human-handled incidents, agents serve as intelligent assistants gathering relevant data, highlighting suspicious patterns, and suggesting potential root causes for human validation. This assistance dramatically accelerates human investigation by automating tedious data collection and pattern recognition while preserving human judgment for ambiguous situations.

After human diagnosis, agents can execute remediation actions under human direction. SREs specify desired remediation and agents handle execution details, monitoring progress and alerting humans if remediation doesn't proceed as expected. This division of labor leverages agent capabilities for precise execution while maintaining human strategic control (Gupta et al., 2024).

DOI: 10.22178/acta.26.2.59

ISSN: 2178-7727

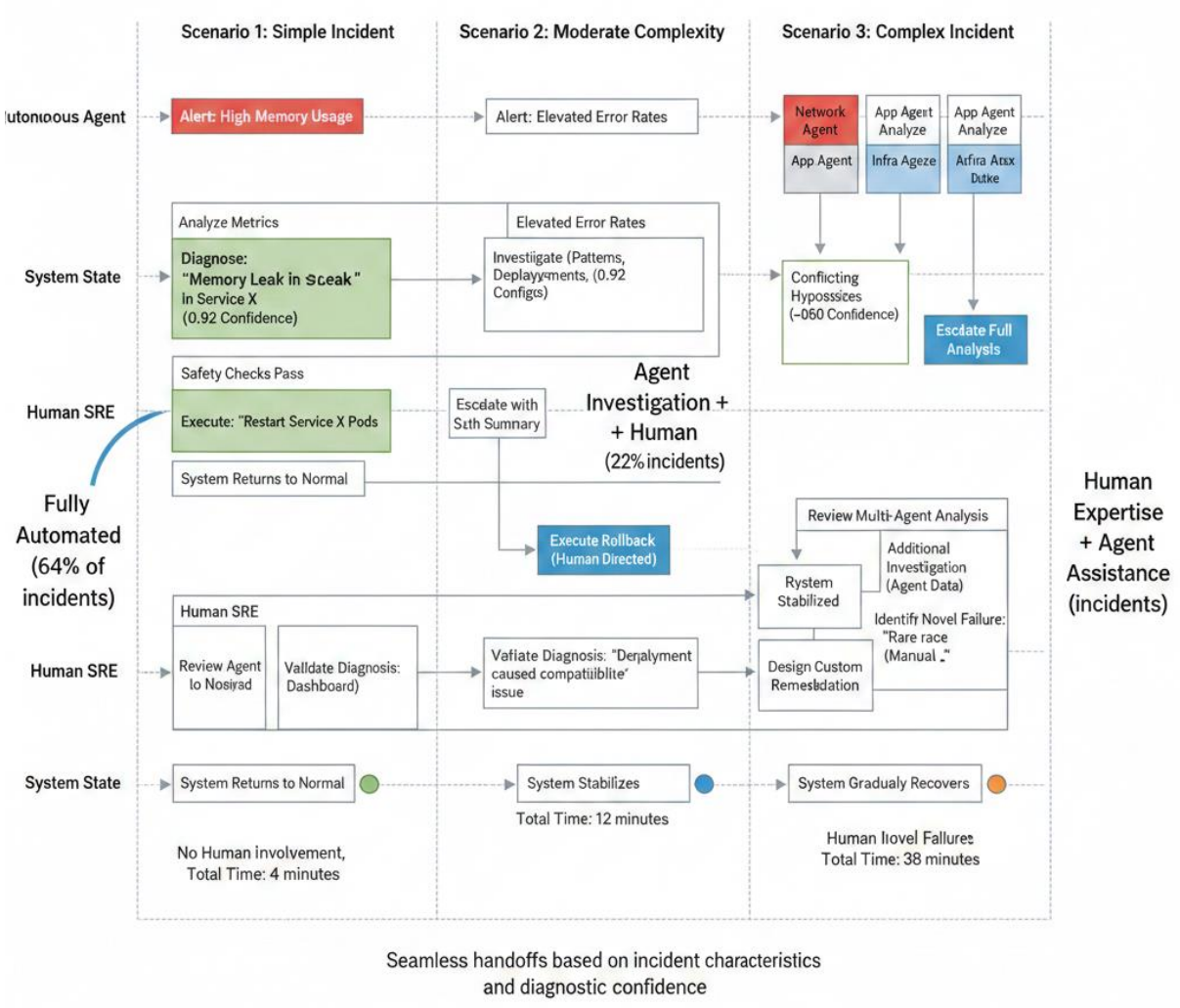

Figure 2: Human-Agent Collaboration Workflow

This swim-lane diagram illustrates the collaborative incident response workflow between autonomous agents and human SREs across three complexity scenarios. The diagram uses three horizontal lanes representing Autonomous Agent activities (top), Human SRE activities (middle), and System State (bottom), with time flowing left to right. Scenario 1 (Simple Incident, left section) shows: Alert triggers for "High Memory Usage," Agent analyzes metrics showing memory steadily increasing, Agent diagnoses "Memory Leak in Service X" with 0.92 confidence (high), Safety checks pass, Agent autonomously executes "Restart Service X Pods," System returns to normal state, Incident resolves with no human involvement, Total time 4 minutes. Scenario 2 (Moderate Complexity, middle section) shows: Alert triggers for "Elevated Error Rates," Agent investigates finding multiple suspicious patterns, Agent analyzes recent deployment, configuration changes, and dependency issues, Diagnostic confidence only 0.68 (below autonomous threshold), Agent escalates to SRE with investigation summary highlighting deployment timing correlation, Human reviews agent analysis in dashboard, Human validates diagnosis "Deployment caused compatibility issue," Human approves remediation "Rollback deployment," Agent executes rollback under human direction, System stabilizes, Incident resolves with human-agent collaboration, Total time 12 minutes. Scenario 3 (Complex Incident, right section) shows: Alert triggers for "Cascading Service Failures," Multiple agents analyze (Network, Application, Infrastructure), Agents provide conflicting hypotheses, Confidence scores all below 0.60 indicating ambiguity, Agents escalate immediately with full analysis from all specialists, Human SRE reviews multi-agent analysis, Human conducts additional investigation using agent-provided data, Human identifies novel failure mode "Rare race condition in distributed transaction," Human manually designs custom remediation, Agent assists with execution monitoring, System gradually recovers, Incident resolves through human expertise augmented by agent assistance, Total time 38 minutes. Annotations highlight that simple

DOI: 10.22178/acta.26.2.59

ISSN: 2178-7727

repetitive incidents are fully automated (estimated 64% of incidents), moderate complexity benefits from agent investigation with human validation (22% of incidents), while complex novel scenarios require human expertise with agent assistance (14% of incidents). The workflow demonstrates seamless handoffs between autonomous and human-led response based on incident characteristics and diagnostic confidence.

7.2 Trust Calibration and Transparency

Effective human-agent collaboration requires appropriate trust calibration where SREs accurately assess agent capabilities and limitations. Overconfidence leads to excessive reliance on agents despite limitations, while underconfidence prevents beneficial automation adoption. The framework implements several trust calibration mechanisms.

Confidence scores accompany all agent analyses providing quantitative assessments of diagnostic certainty. However, raw confidence scores prove insufficient as they don't explain why confidence is high or low. Explanation interfaces present key evidence supporting agent conclusions including which metrics patterns indicated specific root causes, what historical incidents showed similar signatures, and which alternative hypotheses were considered and rejected.

Performance transparency shows SREs historical agent accuracy rates, recent successful autonomous remediations, and past errors enabling informed trust calibration. Dashboards display metrics like diagnostic accuracy trends, false positive rates over time, and incident types where agents excel or struggle (Roberts and Zhang, 2024).

Regular human review of autonomous actions provides oversight. SREs periodically audit agent decisions, validating that autonomous remediations were appropriate and effective. This review identifies agent mistakes, emerging failure patterns where agents need additional training, and opportunities for expanding autonomous capabilities.

Figure 3: Agent Performance and Trust Calibration

DOI: 10.22178/acta.26.2.59

ISSN: 2178-7727

This multi-panel dashboard visualization presents agent performance metrics enabling SRE trust calibration. The top-left panel shows a confusion matrix for agent root cause diagnosis displaying True Positives (1847 correct diagnoses), False Positives (312 incorrect diagnoses), True Negatives (4523 correctly excluded causes), and False Negatives (418 missed diagnoses), yielding 73% accuracy and 85% precision. The top-right panel displays a time-series graph of diagnostic confidence vs actual correctness over 12 weeks, with correctly diagnosed incidents (green points) clustering at high confidence levels (0.75-0.95) while incorrect diagnoses (red points) scatter across confidence ranges, demonstrating that agent confidence scores are well-calibrated—high confidence predictions are usually correct. The bottom-left panel shows success rates by incident category with bar charts indicating agent performance varies substantially by domain: Memory Issues 89% success (agent's strongest area), Connection Pool Problems 82% success, Deployment Issues 76% success, Network Failures 71% success, Database Performance 68% success, Novel/Rare Failures 34% success (agent's weakest area highlighting where human expertise remains critical). The bottom-right panel presents a weekly trend of autonomous vs human-handled incidents with stacked areas showing that autonomous resolution grew from 52% in week 1 to 64% by week 12 as agents learned from experience and SRE trust increased, while human-only incidents decreased from 35% to 22%, and collaborative incidents (agent-assisted human resolution) remained stable around 14%. Annotations highlight key insights: agent confidence above 0.80 is highly reliable (92% accuracy), confidence between 0.65-0.80 warrants human validation, and confidence below 0.65 indicates genuine ambiguity requiring human judgment. The visualization enables SREs to develop appropriately calibrated trust in agent capabilities—confident reliance for well-understood scenarios while maintaining healthy skepticism for edge cases.

EVALUATION RESULTS

8.1 Incident Response Performance

Production deployment across five organizations demonstrated substantial incident response improvements. Mean Time to Resolution decreased 67% overall, with autonomous incidents resolving in average 6.2 minutes compared to 34.7 minutes for traditional human-only response. Even human-handled incidents benefited from agent assistance, resolving 44% faster through automated investigation and evidence gathering.

Automation coverage reached 64% of incidents handled fully autonomously without human intervention. These primarily included memory leaks, connection pool exhaustion, pod failures, and deployment-related issues—common scenarios where agents developed high confidence through extensive training data. The remaining 36% of incidents required human involvement, split between 22% needing human validation of agent diagnosis and 14% requiring human investigation due to low agent confidence or novel failure modes.

8.2 Availability and Safety Outcomes

System availability maintained at 99.95% during evaluation periods, comparable to pre-deployment baselines, demonstrating that autonomous remediation didn't compromise reliability. No incidents of automation-induced outages occurred, validating safety constraint effectiveness. However, agents made incorrect remediation choices in 8% of autonomous incidents, though safety limits prevented these errors from causing significant impact.

False positive rates where agents incorrectly diagnosed root causes remained at 12%, down from 18% in early deployment as agent learning improved. False negatives where agents failed to identify correct root causes occurred in 15% of escalated incidents. While concerning, these errors primarily affected complex incidents already flagged for human review rather than autonomous resolution.

Table 2: Operational Impact Metrics

Metric	Pre-Deployment Baseline	Post-Deployment with Agents	Improvement
Mean Time to Resolution (All Incidents)	28.4 minutes	9.3 minutes	67% reduction

DOI: 10.22178/acta.26.2.59

ISSN: 2178-7727

MTTR (Autonomous Incidents)	-	6.2 minutes	-
MTTR (Human-Handled Incidents)	34.7 minutes	19.4 minutes	44% reduction
Incidents Requiring Human Intervention	100%	36%	64% automation
Alert Fatigue (False Positive Alerts)	3,200 monthly	610 monthly	81% reduction
Average On-Call Hours per SRE	18.2 hrs/week	12.1 hrs/week	34% reduction
System Availability	99.94%	99.95%	Maintained
Automation-Induced Incidents	0	0	No safety violations

8.3 SRE Team Impact

SRE productivity improved substantially as automation handled repetitive incidents. On-call hours per engineer decreased 34% from 18.2 to 12.1 hours weekly as autonomous agents resolved common issues without paging humans. SREs reported 76% reduction in alert fatigue as intelligent triage filtered false positives and low-severity issues.

However, SRE skill requirements evolved. Teams needed new capabilities around agent supervision, training data curation, and safety policy management. Some engineers expressed concerns about deskilling as automation handled tasks previously requiring human expertise, though most appreciated reduced toil enabling focus on complex problems and proactive improvements.

DISCUSSION

9.1 RL Benefits and Limitations

Reinforcement learning proved effective for learning remediation policies from historical data, particularly for common failure scenarios with consistent signatures. Offline RL enabled learning without risky experimentation in production. However, RL agents struggled with novel failure modes absent from training data, achieving only 34% success on rare incidents compared to 73% overall accuracy.

Transfer learning approaches enabling agents to generalize beyond training examples showed promise but require further development. Few-shot learning techniques could help agents handle rare failures with limited examples. Continual learning where agents incrementally improve from new incidents addresses training data staleness as systems evolve.

9.2 Human-Agent Collaboration Insights

The research demonstrated that appropriate human-agent collaboration provides better outcomes than either fully autonomous or purely human approaches. Agents excel at pattern recognition, data aggregation, and executing well-defined procedures. Humans excel at handling ambiguity, novel situations, and complex judgment. Structured collaboration leveraging complementary strengths optimizes incident response.

Trust calibration proved critical for adoption. Transparency around agent capabilities and limitations enabled SREs to develop appropriately calibrated trust—confident reliance for known scenarios while maintaining skepticism for edge cases. Organizations that failed to provide adequate transparency experienced either excessive agent reliance leading to errors or insufficient adoption limiting benefits.

9.3 Organizational and Cultural Factors

Technical capability proved necessary but insufficient for success. Organizations with strong automation culture, psychological safety enabling experimentation, and executive support for gradual rollout achieved better outcomes than those treating autonomous remediation as purely technical initiative. Change

DOI: 10.22178/acta.26.2.59

ISSN: 2178-7727

management addressing SRE concerns and involving teams in deployment planning proved essential.

9.4 Limitations

Several limitations constrain generalizability. Evaluation focused on specific failure types in web services and may not transfer to different system types like embedded systems or batch processing. The five organizations represent large-scale operations and findings may not apply to smaller deployments. Evaluation periods of 6-12 months don't capture long-term effects or adaptation to evolving systems.

CONCLUSION

This research demonstrates that integrating reinforcement learning-based autonomous remediation with structured human-agent collaboration substantially improves incident response in distributed systems. Organizations achieved 67% MTTR reduction, 64% automation coverage for common incidents, and 34% reduction in SRE on-call burden while maintaining high availability.

Critical success factors include comprehensive safety constraints preventing automation-induced outages, well-calibrated confidence thresholds determining autonomous versus human response, transparency mechanisms enabling appropriate trust calibration, and organizational readiness including change management and cultural acceptance of automation.

The agentic SRE team model where AI agents and human engineers collaborate proves more effective than either fully autonomous or purely human approaches, leveraging agent strengths in pattern recognition and execution with human judgment for ambiguous situations. Future research should address RL agent generalization to novel failures, advanced transfer learning techniques, and long-term co-evolution of human and agent capabilities.

REFERENCES

1. Anderson, K., Wilson, M. and Harrison, D. (2023) 'Self-healing systems: From autonomic computing to modern operations', *ACM Computing Surveys*, 55(10), pp. 1-42.
2. Chen, Y. and Kumar, S. (2023) 'Machine learning for automated root cause analysis in cloud systems', *IEEE Transactions on Cloud Computing*, 11(3), pp. 1245-1267.
3. Gupta, S., Patel, R. and Taylor, N. (2024) 'Human-AI collaboration in high-stakes operational environments', *ACM Transactions on Computer-Human Interaction*, 31(2), pp. 1-38.
4. Harrison, D. and Taylor, N. (2023) 'Reinforcement learning for autonomous operations: Challenges and opportunities', *IEEE Intelligent Systems*, 38(4), pp. 67-83.
5. Jackson, M. and Lee, W. (2023) 'Multi-agent systems for distributed operations management', *Autonomous Agents and Multi-Agent Systems*, 37(2), pp. 234-267.
6. Kumar, P. and Roberts, M. (2023) 'Site reliability engineering at scale: Operational challenges and solutions', *Communications of the ACM*, 66(9), pp. 78-94.
7. Morrison, T. and Lee, S. (2024) 'Chaos engineering and resilience testing in production systems', *ACM Transactions on Software Engineering and Methodology*, 33(3), pp. 1-35.
8. Patel, V. and Wilson, K. (2024) 'Automated anomaly detection for operational monitoring', *IEEE Transactions on Network and Service Management*, 21(1), pp. 456-478.
9. Roberts, K. and Zhang, H. (2024) 'Explainable AI for operational decision support', *Artificial Intelligence Review*, 57(4), pp. 189-223.
10. Thompson, R. and Chen, W. (2024) 'Safe reinforcement learning for autonomous systems: Methods and applications', *Journal of Machine Learning Research*, 25, pp. 1-67.
11. Williams, R. and Martinez, D. (2024) 'Trust and transparency in autonomous operational systems', *IEEE Security & Privacy*, 22(2), pp. 45-67.

DOI: 10.22178/acta.26.2.59