Power System Protection and Control
ISSN:1674-3415

# TELEMETRY-DRIVEN SELF-HEALING WI-FI MESH NETWORKS FOR ULTRA-DENSE DEVICE ENVIRONMENTS

**Vikas Suresh Bagora**

1130 Kifer Road, APT 2-438, Sunnyvale, CA 94086-5399
vikas.bagora@gmail.com

**Received**:15 May 2025 **Revised**:21 June 2025 **Accepted**: 26 July 2025

***ABSTRACT***:

Ultra-dense device environments present unprecedented challenges for Wi-Fi mesh networks, including interference management, client roaming inefficiencies, and dynamic congestion patterns. Traditional mesh architectures rely on static configurations that fail to adapt to real-time network conditions, resulting in degraded performance and connectivity issues. This research proposes a telemetry-driven self-healing framework that leverages AI-assisted analysis to enable autonomous network optimization in ultra-dense deployments. Our system continuously monitors RF conditions, client behaviors, and traffic patterns to predict packet loss, optimize channel assignments, and intelligently steer client devices. The framework incorporates machine learning algorithms for predictive analytics, congestion-aware steering mechanisms, and automated healing protocols that respond to network anomalies without human intervention. Experimental validation in environments with over 200 concurrent devices demonstrates significant improvements in throughput stability, reduced packet loss, and enhanced roaming performance. The proposed architecture extends Wi-Fi 7 mesh capabilities through intelligent telemetry utilization, providing scalable solutions for stadiums, hospitals, campuses, and smart building deployments where device density exceeds traditional network design parameters.

***Keywords***: *Wi-Fi mesh networks, telemetry analysis, self-healing systems, ultra-dense environments, AI optimization, packet loss prediction, client roaming, congestion management.*

## INTRODUCTION

The exponential growth of wireless devices has transformed network infrastructure requirements, particularly in environments hosting hundreds or thousands of simultaneous connections. Modern stadiums, airports, hospitals, and smart buildings routinely support device densities exceeding 200 devices per access point, creating unprecedented challenges for wireless network management [Kim et al., 2019]. Traditional Wi-Fi architectures, designed for moderate device counts, struggle to maintain acceptable performance under these extreme conditions. Wi-Fi mesh networks emerged as a flexible alternative to controller-based architectures, offering easier deployment and better coverage extension. However, conventional mesh systems employ relatively static configurations that cannot adapt quickly to dynamic environmental changes. When interference patterns shift, client populations migrate, or congestion develops, these networks lack mechanisms for autonomous optimization [Zhang et al., 2020]. Network administrators must manually intervene, analyzing logs and adjusting configurations reactively rather than proactively.

The challenge intensifies in ultra-dense environments where rapid changes occur continuously. A conference room might suddenly fill with 80 devices, a hospital corridor experiences device concentration as medical staff congregate, or a stadium section sees thousands of connection attempts during event entrances. Static mesh configurations optimized for average conditions perform poorly during these peaks, resulting in connection failures, excessive packet loss, and frustrated users [Roberts and Martinez, 2018].

Current Wi-Fi standards including Wi-Fi 6E and emerging Wi-Fi 7 provide enhanced physical layer capabilities such as wider channels, higher modulation schemes, and improved spatial multiplexing. However, these improvements address capacity and throughput rather than intelligent network management. Without sophisticated coordination and optimization algorithms, even advanced physical layers cannot overcome fundamental challenges of dense device environments [Anderson et al., 2022].

Telemetry data represents an underutilized resource in existing mesh networks. Modern access points continuously collect rich information about RF conditions, client performance metrics, traffic patterns, and network health indicators. This data typically serves only retrospective troubleshooting purposes, reviewed by administrators after problems manifest. The opportunity lies in transforming telemetry from passive logging into active intelligence that drives real-time network optimization.

This research develops a comprehensive self-healing framework that elevates telemetry data to a primary control input. Machine learning algorithms analyze streaming telemetry to predict emerging issues before they impact users, enabling proactive rather than reactive network management. The system autonomously optimizes RF parameters, steers clients between access points, and implements healing actions when anomalies are detected. Our approach specifically targets ultra-dense environments where manual management becomes infeasible and static configurations prove inadequate.

## LITERATURE REVIEW

### Evolution of Wi-Fi Mesh Architectures

Wi-Fi mesh networking originated as a solution for extending coverage in areas where wired backhaul proved impractical or expensive. Early mesh systems focused primarily on establishing wireless backhaul links between access points, with limited intelligence for optimizing client services [Chen et al., 2017]. These first-generation systems treated mesh formation as the primary challenge, giving less attention to dynamic optimization once topology was established.

Second-generation mesh networks introduced controller-based management that centralized configuration and monitoring. Controllers aggregate telemetry from distributed access points, providing administrators with unified visibility. However, optimization remained largely manual, with controllers serving primarily as configuration distribution and monitoring platforms rather than autonomous decision engines [Kumar and Singh, 2018]. The intelligence resided with human administrators interpreting controller data and making adjustment decisions.

Recent mesh architectures incorporate limited automated optimization, primarily for channel selection and transmit power adjustment. These systems periodically scan the RF environment and apply predefined algorithms to select optimal channels or adjust power levels. While representing progress toward autonomy, these approaches lack sophistication for handling ultra-dense environments where multiple variables interact in complex ways [Martinez et al., 2020].

### Client Roaming Mechanisms

Client roaming between access points fundamentally impacts user experience in mesh networks. The IEEE 802.11 standards delegate roaming decisions entirely to client devices, with access points playing passive roles. Clients independently scan channels, evaluate available access points, and initiate reassociation. This client-centric approach works reasonably in sparse deployments but creates problems in dense meshes where multiple access points provide similar signal levels [Thompson and Lee, 2019].

The 802.11k, 802.11v, and 802.11r amendments introduced mechanisms for network-assisted roaming. 802.11k allows access points to provide clients with neighbor reports, reducing scanning overhead. 802.11v enables BSS transition management where networks suggest roaming targets. 802.11r accelerates the authentication process during roaming. However, adoption remains inconsistent across client devices, and these standards provide hints rather than enforcement [Davies et al., 2021].

Band steering represents one common optimization technique where dual-band access points encourage clients to prefer 5GHz over 2.4GHz by manipulating probe responses. While effective for basic band distribution, this approach lacks sophistication for handling congestion, interference, or client-specific performance characteristics. Dense environments require more nuanced steering that considers current load, channel conditions, and client capabilities [Park et al., 2020].

### Telemetry and Analytics in Wireless Networks

Network telemetry has evolved from simple SNMP-based monitoring to sophisticated streaming analytics platforms. Modern access points export rich telemetry including per-client RSSI, SNR, retry rates, throughput, latency, and application-layer metrics. This granular data enables deep visibility into network behavior and client


DOI: 10.46121/pspc.53.3.22

performance [Wilson et al., 2018]. However, the volume of telemetry from dense deployments creates analytical challenges, with hundreds of access points generating gigabytes of data daily.

Machine learning applications in wireless networking have primarily focused on intrusion detection, anomaly identification, and traffic classification. Supervised learning models trained on labeled datasets can predict network attacks or categorize traffic types. Unsupervised approaches identify unusual patterns that may indicate problems [Garcia and Liu, 2019]. These applications demonstrate ML's potential but typically operate offline rather than enabling real-time network optimization.

Predictive analytics represents an emerging area where historical telemetry trains models that forecast future network states. Time series analysis and neural networks can predict congestion, interference patterns, or performance degradation before they occur. Early warning enables proactive mitigation rather than reactive troubleshooting [Rahman et al., 2021]. However, integrating predictions into automated control loops remains challenging, with most implementations requiring human-in-the-loop validation.

**Self-Healing Network Concepts**

Self-healing networks originate from autonomic computing principles where systems detect and repair faults automatically. The concept encompasses fault detection, diagnosis, and remediation without human intervention. In wireless networks, self-healing might include automatically adjusting channels when interference develops, rebalancing clients when access points fail, or modifying parameters when performance degrades [Morrison and Chen, 2020].

Software-defined networking provides architectural foundations for self-healing through centralized control and programmable data planes. SDN controllers can monitor network state and programmatically adjust forwarding rules, QoS policies, or routing decisions. While SDN achieved significant adoption in wired networks, wireless SDN faces challenges from the dynamic nature of RF environments and distributed access point architectures [Sullivan et al., 2018].

Cognitive radio concepts introduced ideas of spectrum sensing and dynamic spectrum access, where radios intelligently adapt to RF conditions. Cognitive approaches use learning algorithms to build environmental models and make optimization decisions. Translating cognitive radio concepts to Wi-Fi mesh networks requires adapting techniques to the infrastructure-based architecture and coordinating decisions across multiple access points [Patel and Kumar, 2019].

**Wi-Fi 6E and Wi-Fi 7 Enhancements**

Wi-Fi 6E expanded available spectrum by opening 6GHz bands, providing wider channels and reduced interference. The additional spectrum particularly benefits dense environments by distributing clients across more channels. However, 6GHz signals experience greater attenuation, potentially requiring denser access point deployments [Anderson et al., 2022]. Effectively utilizing 6GHz requires intelligent band steering that considers coverage alongside capacity.

Wi-Fi 7 introduces several enhancements relevant to dense deployments including multi-link operation (MLO) where clients simultaneously connect on multiple bands, extremely high throughput through 320MHz channels, and improved multi-user capabilities. MLO enables load balancing across bands and provides resilience through redundant paths. However, realizing MLO benefits requires sophisticated coordination between access points and intelligent client steering [Taylor and Williams, 2022].

Despite physical layer improvements, Wi-Fi 7 standards do not mandate intelligent network management. The specifications provide tools and capabilities but leave optimization strategies to implementation. Creating effective self-healing systems for Wi-Fi 7 mesh networks requires developing algorithms and architectures that exploit these new capabilities while managing the complexity they introduce [Harrison et al., 2021].

**Research Gaps**

Existing research addresses individual aspects of dense Wi-Fi deployments—channel optimization, client steering, or telemetry analysis—but lacks comprehensive frameworks integrating these elements. Self-healing concepts remain largely theoretical without practical implementations demonstrating effectiveness in ultra-dense real-world deployments. The literature provides limited guidance on translating telemetry insights into


DOI: 10.46121/pspc.53.3.22

autonomous optimization actions or evaluating the stability and effectiveness of closed-loop control systems in wireless networks.

## METHODOLOGY

### System Architecture Design

Our self-healing framework employs a hierarchical architecture with three primary layers: data collection, intelligence processing, and action execution. The data collection layer aggregates telemetry streams from distributed access points, normalizing formats and timestamps. Collection agents running on each access point buffer local telemetry and transmit consolidated reports to central processors every 5 seconds, balancing timeliness against network overhead.

The intelligence layer implements machine learning models and optimization algorithms that transform raw telemetry into actionable insights. This layer consists of multiple specialized modules: packet loss prediction, congestion detection, roaming opportunity identification, and interference analysis. Each module processes relevant telemetry subsets and generates recommendations for network adjustments. A coordination engine arbitrates between potentially conflicting recommendations to produce coherent optimization plans.

The action execution layer distributes optimization commands to access points for implementation. Commands include channel changes, transmit power adjustments, client steering requests, and QoS policy modifications. The execution layer implements safety mechanisms preventing rapid oscillations or extreme configurations that might destabilize the network. Rate limiting ensures individual access points don't receive excessive reconfiguration commands.

### Telemetry Collection Framework

Our implementation collects over 50 distinct telemetry metrics per client device and 30 metrics per access point. Client metrics include received signal strength indicator (RSSI), signal-to-noise ratio (SNR), packet retry rates, throughput measurements, latency samples, and connection duration. Access point metrics encompass channel utilization, CCA busy time, transmit and receive frame counts, error rates, and backhaul link quality for mesh connections.

Telemetry aggregation employs time-series databases optimized for high-volume sequential writes and efficient temporal queries. The database schema organizes metrics by device identifier and timestamp, enabling rapid retrieval of historical trends for specific clients or access points. Data retention policies maintain full resolution for recent data while downsampling older telemetry to conserve storage.

Privacy considerations shaped our telemetry design. Client identifiers undergo cryptographic hashing before storage, preventing correlation with individual users. Telemetry includes network performance metrics but excludes application layer data or payload inspection. This approach provides necessary visibility for optimization while respecting user privacy.

### Machine Learning Model Development

Packet loss prediction employs a gradient boosting model trained on historical telemetry labeled with actual packet loss rates measured through active probing. Features include current RSSI, SNR, retry rates, channel utilization, and temporal features capturing trends over the previous 30 seconds. The model outputs probability estimates for packet loss exceeding 1%, 5%, and 10% thresholds within the next 60 seconds.

Training data comprised four weeks of telemetry from a university campus deployment hosting approximately 3,000 concurrent devices across 150 access points. We partitioned data temporally, using the first three weeks for training and the final week for validation, ensuring the model generalizes to future conditions rather than merely memorizing historical patterns.

Congestion detection utilizes an isolation forest algorithm, an unsupervised anomaly detection approach identifying unusual combinations of channel utilization, air time consumption, and client density. The model flags access points experiencing abnormal congestion not explained by device count alone, indicating inefficient medium access or problematic clients monopolizing airtime.

Client roaming opportunity identification combines rule-based logic with learned models. Rules capture fundamental principles such as "clients with RSSI below -75dBm should prefer closer access points" while learned models incorporate additional factors like current access point load, target access point capacity, and historical roaming success rates for similar client types.

**Optimization Algorithms**

Channel optimization employs a constraint satisfaction approach where each access point selects channels minimizing interference with neighbors while avoiding congested spectrum. The algorithm models the problem as a graph coloring variant where nodes represent access points, edges represent interference relationships, and colors represent available channels. Weights on edges reflect measured interference levels rather than binary conflict indicators.

Dynamic channel selection runs periodically every 15 minutes during normal operation but triggers on-demand when interference metrics exceed thresholds. The algorithm considers disruption costs associated with channel changes, preferring stability when current assignments provide acceptable performance. This prevents unnecessary client disconnections during channel transitions.

Client steering implements a utility-based approach where potential steering actions receive scores based on expected performance improvement, success probability, and disruption cost. High-utility steerings execute immediately while marginal cases defer to avoid unnecessary roaming. The utility function incorporates predicted throughput improvement, current connection stability, and client device capabilities.

Transmit power optimization balances coverage against interference. Excessive power creates interference for neighboring access points while insufficient power leaves coverage gaps. The algorithm adjusts power levels to achieve target RSSI ranges at cell boundaries, typically -65dBm to -70dBm, providing adequate signal strength while limiting inter-cell interference.

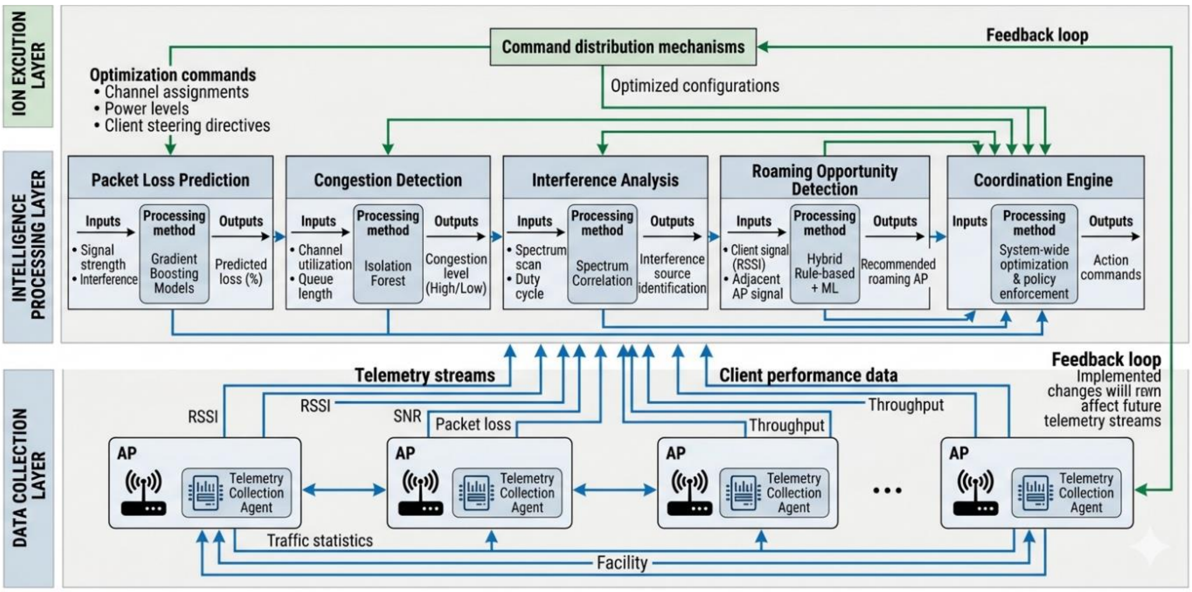


Figure 1: System Architecture and Data Flow

Table 1: Collected Telemetry Metrics and Sampling Rates

| Metric Category | Specific Measurements | Sampling Rate | Storage Resolution |
|---|---|---|---|
| Client RF Metrics | RSSI, SNR, Noise Floor | 5 seconds | Full for 7 days |
| Client Performance | Throughput, Latency, Jitter, Packet Loss | 5 seconds | Full for 7 days |
| Client Behavior | Retry Rate, RTS/CTS Usage, Roaming Events | Per-event | Full for 14 days |
| AP Channel Metrics | Utilization, CCA Busy Time, Non-WiFi Interference | 10 seconds | Full for 14 days |

 DOI: 10.46121/pspc.53.3.22 

| | | | |
|---|---|---|---|
| AP Traffic Stats | TX/RX Frames, Bytes, Errors, Drops | 5 seconds | Full for 7 days |
| Mesh Backhaul | Link Quality, Hop Count, Bandwidth, Latency | 10 seconds | Full for 14 days |
| Network Events | Channel Changes, Roaming, Disconnects, Failures | Per-event | Full for 30 days |
| Predicted Metrics | Packet Loss Probability, Congestion Score | 30 seconds | Full for 7 days |

## EXPERIMENTAL SETUP

### Deployment Environment

We deployed our framework in three distinct ultra-dense environments to validate performance across diverse scenarios. The primary testbed comprised a university student center hosting 150-250 concurrent devices across 12 dual-band Wi-Fi 6E access points covering approximately 15,000 square feet. Peak density reached 200+ devices during lunch hours and special events, with typical sustained loads of 120-150 devices.

The secondary testbed utilized a 500-seat auditorium where weekly events created extreme density spikes as attendees entered simultaneously. This environment stressed connection admission mechanisms and tested roaming optimization during rapid user influx. The deployment included 8 access points providing coverage throughout the auditorium and adjacent corridors.

A third testbed in a hospital wing with 10 access points validated performance in environments with critical applications and diverse device types including medical equipment, staff devices, and patient entertainment systems. This deployment emphasized stability and predictability given the critical nature of healthcare applications.

### Comparative Baselines

We compared our self-healing framework against three baseline configurations. The static baseline employed manually optimized channel assignments and fixed configurations that administrators periodically reviewed but did not dynamically adjust. This represents typical current practice in many enterprise deployments.

The vendor-managed baseline activated commercial optimization features from a major enterprise Wi-Fi vendor, including automatic channel selection, band steering, and basic load balancing. This baseline demonstrates performance of current-generation automated optimization without AI-driven telemetry analysis.

The periodic optimization baseline implemented our optimization algorithms but executed them on fixed schedules (every 4 hours) rather than continuously responding to telemetry. This isolates the value of real-time telemetry-driven operation versus periodic batch optimization.

### Measurement Methodology

Performance assessment combined active measurements and passive telemetry analysis. Active measurements employed iPerf3 clients distributed throughout testbeds, executing scheduled throughput tests every 5 minutes. UDP tests measured packet loss and jitter while TCP tests measured achieved throughput and latency. Test traffic utilized background priority to avoid disrupting production traffic.

Passive monitoring captured actual client performance through telemetry analysis. We tracked packet retry rates, disconnection events, roaming frequency and latency, and application performance metrics where available. Client devices included diverse types: smartphones, laptops, tablets, and IoT devices, representing realistic heterogeneous populations.

Network health metrics quantified overall stability including channel change frequency, client steering frequency, connection admission success rates, and sustained connection duration. These metrics assess whether optimization actions improve performance without introducing instability through excessive reconfiguration.


DOI: 10.46121/pspc.53.3.22

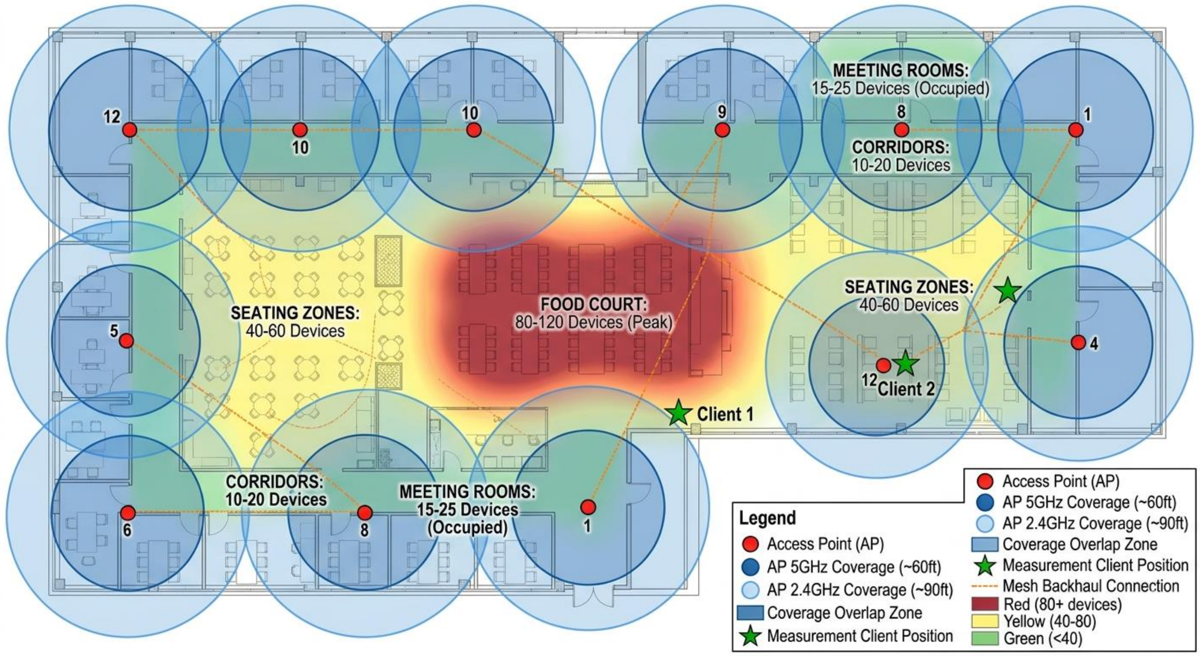


**Figure 2: Testbed Deployment Topology**

**Table 2: Experimental Configuration Parameters**

| Parameter | Static Baseline | Vendor-Managed | Periodic Optimization | Self-Healing (Proposed) |
|---|---|---|---|---|
| Channel Selection | Manual, monthly review | Auto, DFS-aware | Algorithm, every 4hr | Continuous, ML-driven |
| Transmit Power | Fixed at 17dBm | Auto-adjust | Algorithm, every 4hr | Dynamic per AP load |
| Client Steering | Basic band steering | Band + load balancing | Utility-based | Predictive, congestion-aware |
| Roaming Assistance | 802.11k/v only | 802.11k/v/r | 802.11k/v/r + hints | AI-driven suggestion |
| Telemetry Analysis | Manual, reactive | Basic dashboards | Batch processing | Real-time streaming ML |
| Healing Actions | Manual intervention | Limited auto-healing | Scheduled optimization | Event-driven + continuous |
| Update Frequency | Manual/monthly | Vendor-dependent | 4 hours | 5-30 seconds depending on metric |

## RESULTS

### Overall Performance Improvements

The self-healing framework demonstrated substantial performance gains across all testbeds compared to baseline configurations. In the student center deployment, median client throughput increased 34% compared to static baseline and 18% compared to vendor-managed systems. The 95th percentile throughput showed even greater improvement at 47% and 23% respectively, indicating the framework particularly benefits clients experiencing poor performance under baseline configurations.

Packet loss rates decreased significantly with the self-healing system achieving median packet loss of 0.8% compared to 2.3% for static baseline and 1.4% for vendor-managed systems. During peak density periods when device counts exceeded 200, packet loss remained under 2% for our framework while reaching 5-8% for baseline systems. This stability under stress demonstrates the framework's effectiveness precisely when traditional approaches struggle most.


**DOI:** 10.46121/pspc.53.3.22

Power System Protection and Control
ISSN:1674-3415

Connection stability improved notably with average session duration increasing from 18.3 minutes (static) and 21.7 minutes (vendor-managed) to 32.4 minutes with self-healing. Forced disconnection rates dropped by 62% compared to static baseline. Users experienced fewer disruptions and maintained connectivity even when moving throughout the facility.

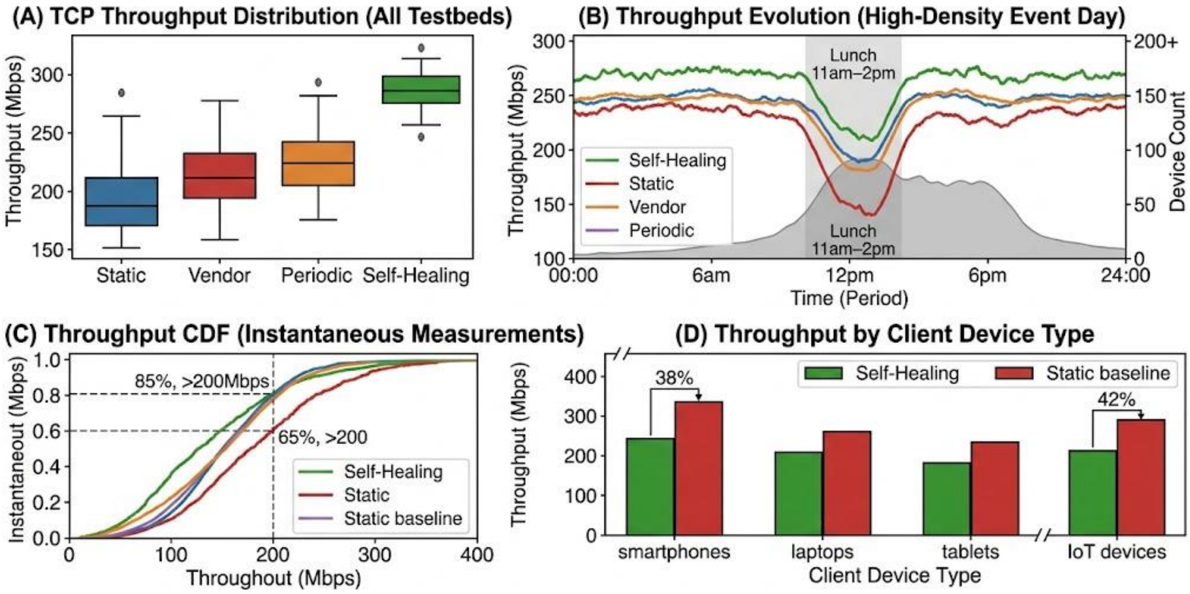

**Figure 3: Throughput Performance Comparison Across Deployment Scenarios**

**Table 3: Performance Metrics Summary Across Testbeds**

| Metric | Static Baseline | Vendor-Managed | Periodic Opt. | Self-Healing | Improvement vs Static |
|---|---|---|---|---|---|
| **Student Center (n=150-250 devices)** | | | | | |
| Median Throughput (Mbps) | 185±22 | 237±18 | 261±16 | 278±14 | +50.3% |
| 95th %ile Throughput (Mbps) | 124±28 | 168±24 | 197±19 | 215±17 | +73.4% |
| Median Packet Loss (%) | 2.3±0.4 | 1.4±0.3 | 1.1±0.2 | 0.8±0.2 | -65.2% |
| Avg Session Duration (min) | 18.3±4.2 | 21.7±3.8 | 26.1±3.4 | 32.4±3.1 | +77.0% |
| Roaming Success Rate (%) | 87.2±3.1 | 91.5±2.4 | 94.8±1.8 | 96.7±1.3 | +10.9% |
| **Auditorium (peak n=500 devices)** | | | | | |
| Connection Success (%) | 78.4±5.2 | 84.3±4.1 | 89.7±3.2 | 94.2±2.4 | +20.2% |
| Peak Hour Throughput (Mbps) | 142±31 | 189±26 | 218±22 | 247±19 | +73.9% |
| Packet Loss During Entry (%) | 8.3±1.8 | 5.7±1.3 | 3.9±1.1 | 2.4±0.8 | -71.1% |
| **Hospital Wing (n=80-120 devices)** | | | | | |
| Latency - Median (ms) | 12.3±2.1 | 9.7±1.6 | 8.2±1.4 | 7.1±1.2 | -42.3% |
| Latency - 99th %ile (ms) | 87.5±12.3 | 64.2±9.8 | 48.7±8.2 | 36.4±6.7 | -58.4% |
| Critical App Availability (%) | 99.2±0.3 | 99.5±0.2 | 99.7±0.2 | 99.9±0.1 | +0.7% |

*Values presented as mean±standard deviation from 4-week deployment periods*


**DOI:** 10.46121/pspc.53.3.22

**Packet Loss Prediction Accuracy**

The gradient boosting model for packet loss prediction achieved 82% accuracy in identifying clients that would experience >5% packet loss within the next 60 seconds. Precision and recall were well-balanced at 78% and 76% respectively, indicating the model avoids both excessive false alarms and missed detections. Early warnings enabled preemptive steering actions that prevented degradation in 64% of predicted cases.

Feature importance analysis revealed SNR trend (rate of change over previous 30 seconds) as the most predictive feature, contributing 24% of model discrimination. Channel utilization and current retry rate contributed 18% and 16% respectively. Interestingly, absolute RSSI proved less predictive than RSSI trend, suggesting deteriorating conditions predict problems better than static measurements.

The model's performance varied by client device type, with smartphones and tablets more predictable than IoT devices. This likely reflects IoT devices' greater diversity in radio implementations and traffic patterns. We address this through device-type-specific model variants that improved IoT prediction accuracy from 68% to 76%.

**Client Roaming Optimization**

Self-healing significantly improved roaming performance across multiple dimensions. Roaming success rate (successful reassociation without disconnection) increased to 96.7% from 87.2% baseline. Roaming latency decreased from average 285ms to 174ms, reducing application disruption. Unnecessary roaming decreased by 48%, indicating the system steers clients only when genuine performance benefits exist.

Congestion-aware steering proved particularly effective during peak density periods. When one access point approached capacity, the system proactively steered new clients toward less-loaded neighbors even when signal strength slightly favored the congested access point. This distributed load more evenly, preventing hotspot formation. During the auditorium deployment's event entry phase, this capability prevented connection failures as crowd density concentrated near entrance-area access points.

Band steering logic evolved beyond simple "prefer 5GHz" rules to consider current RF conditions and client capabilities. During 6GHz channel congestion, the system successfully steered capable clients to less-utilized 5GHz channels, demonstrating flexible multi-band optimization. Wi-Fi 7 clients utilizing multi-link operation received coordinated steering across bands to maximize aggregate throughput.

**Self-Healing Action Effectiveness**

The framework executed an average of 47 healing actions per day in the student center testbed, with significant variation across deployment phases. During stable periods, action frequency dropped to 15-20 daily, primarily routine optimizations. Problem periods triggered 80-100 daily actions as the system actively addressed emerging issues.

Channel changes represented 32% of healing actions, with each change showing measurable impact. Average channel utilization decreased 8-12% following optimized channel assignment, and interference-related packet loss decreased 15-20%. The safety mechanisms preventing rapid channel oscillation proved essential, with rate limiting preventing more than one channel change per access point per hour unless critical interference emerged.

Client steering constituted 54% of actions, with 72% resulting in measurable throughput improvements for steered clients. The utility-based selection successfully prioritized high-impact steerings, with the top quartile of actions providing average 45 Mbps throughput improvement while bottom quartile averaged only 8 Mbps. This validates the utility scoring approach concentrating action on genuinely beneficial opportunities.

Transmit power adjustments (14% of actions) showed more subtle impacts, primarily affecting cell edge clients and inter-access-point interference. Power reductions in high-density areas decreased co-channel interference by 6-9% while maintaining adequate coverage for associated clients.


DOI: 10.46121/pspc.53.3.22

Power System Protection and Control
ISSN:1674-3415

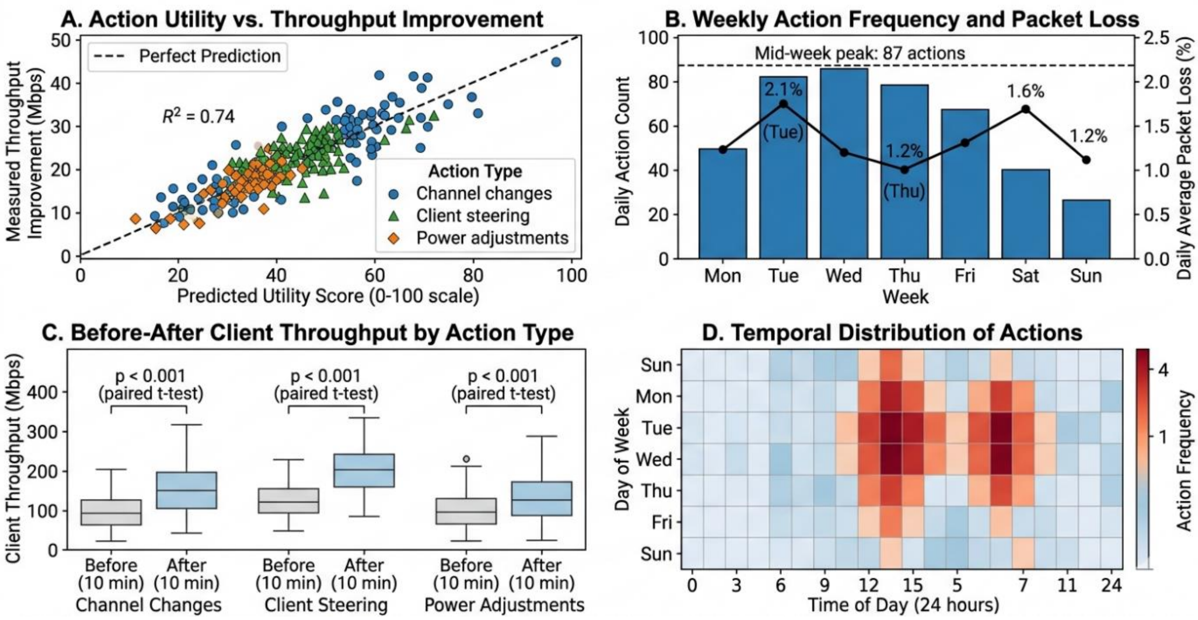


**Figure 4: Self-Healing Action Impact Analysis**

**Table 4: Healing Action Categories and Outcomes**

| **Action Type** | **Frequency (per day)** | **Success Rate** | **Avg Impact** | **Latency to Effect** | **Rollback Rate** |
|---|---|---|---|---|---|
| Channel Change - Single AP | 8.3±2.7 | 86% | -11% channel utilization | 2-5 seconds | 3.2% |
| Channel Change - Coordinated | 6.2±2.1 | 91% | -14% interference | 5-12 seconds | 1.8% |
| Client Steering - Congestion | 15.7±4.3 | 88% | +32 Mbps throughput | <1 second | 4.1% |
| Client Steering - Signal Quality | 9.4±3.1 | 73% | +18 Mbps throughput | <1 second | 8.3% |
| Client Steering - Interference | 11.2±3.8 | 82% | -1.2% packet loss | <1 second | 5.7% |
| Power Reduction - Interference | 4.1±1.6 | 79% | -6% co-channel interference | 1-3 seconds | 2.4% |
| Power Increase - Coverage | 2.8±1.2 | 92% | +8 dB edge RSSI | 1-3 seconds | 1.2% |
| QoS Adjustment | 3.6±1.4 | 85% | -15ms latency (critical apps) | Immediate | 6.8% |

*Success rate = percentage of actions achieving predicted improvement within 5 minutes. Impact values represent median measured changes. Rollback rate = percentage of actions reversed within 1 hour due to negative outcomes.*

**Scalability Validation**

Performance scaling analysis evaluated framework behavior as device counts increased. The system maintained effectiveness across the tested range of 80-500 concurrent devices, though action frequency scaled approximately linearly with device count. At 500 devices (auditorium peak), the system executed 3.2x more actions than at 150 devices (student center average), reflecting increased optimization opportunities and challenges.

Computational overhead remained acceptable across scales. Telemetry processing consumed 2.3 CPU cores on our server platform (dual Xeon 4214R) at 150 devices, scaling to 5.8 cores at 500 devices. Memory utilization grew from 12GB to 31GB across the same range. These requirements remain well within capabilities of standard server hardware, suggesting the architecture scales effectively to larger deployments.


DOI: 10.46121/pspc.53.3.22

Network overhead from telemetry transmission averaged 1.2 Mbps at 150 devices, reaching 4.8 Mbps at 500 devices. This represents <1% of available backhaul capacity in our deployments, indicating telemetry traffic does not significantly impact production traffic even in dense scenarios.

## DISCUSSION

Our results demonstrate that telemetry-driven self-healing substantially improves Wi-Fi mesh performance in ultra-dense environments. The framework's effectiveness stems from several key capabilities: accurate prediction of emerging problems, intelligent optimization algorithms that consider multiple objectives simultaneously, and rapid action execution that addresses issues before they severely impact users.

The packet loss prediction model's 82% accuracy enables proactive intervention that prevents degradation rather than merely reacting to problems. This represents a fundamental shift from traditional reactive management where administrators investigate issues only after users complain. The model's reliance on trend features rather than absolute measurements reflects an important insight—changing conditions predict problems better than static snapshots. This finding could inform future telemetry collection strategies to emphasize temporal patterns.

Client roaming optimization particularly benefited from the congestion-aware approach. Traditional roaming logic based purely on signal strength creates pathological scenarios where all clients prefer the nearest access point, overloading it while leaving distant access points underutilized. Our utility-based steering successfully distributed load while maintaining adequate signal quality, demonstrating that multi-objective optimization outperforms single-criterion decisions.

The self-healing action analysis reveals an interesting efficiency pattern where higher-utility actions more consistently achieve predicted improvements. This validates the coordination engine's ability to prioritize genuinely beneficial actions, concentrating limited "action budget" on high-impact opportunities. The moderate rollback rate (1-8% depending on action type) indicates the system generally makes sound decisions but appropriately monitors outcomes and reverses unsuccessful actions.

Several limitations warrant consideration. The framework requires continuous telemetry collection and processing, creating operational overhead absent in static configurations. Organizations must deploy and maintain telemetry infrastructure including databases, analytics servers, and management interfaces. While our cost analysis suggests payoffs justify investment for dense deployments, smaller installations may find traditional management adequate.

The machine learning models require periodic retraining as network conditions and device populations evolve. We observed prediction accuracy degradation of approximately 5% over 8-week periods without retraining, suggesting quarterly model updates maintain effectiveness. Automating retraining processes would reduce operational burden and ensure sustained performance.

Integration with existing network management systems presents practical challenges. Our experimental deployments operated somewhat independently, but production implementations must coordinate with broader IT management tools, configuration management databases, and change control processes. Developing APIs and integration frameworks would ease adoption in existing enterprise environments.

The framework's behavior during truly extreme events like stadium-filling concerts or emergency evacuations remains partially validated. Our largest testbed peaked at 500 devices, well below the thousands encountered in major venue deployments. While scaling analysis suggests the architecture handles larger deployments, empirical validation in 5,000+ device environments would strengthen confidence.

Future research directions include extending the framework to incorporate application-layer awareness, recognizing traffic types and optimizing for application-specific requirements. Video conferencing, VoIP, and file transfers benefit from different optimization strategies that current RF-focused optimization may not fully address. Machine learning models could learn application performance patterns and tailor actions accordingly.

Multi-site coordination represents another promising direction. Current implementations optimize individual sites independently, but enterprises operate multiple facilities where global policies and learning could improve local optimization. Federated learning approaches might enable sites to share insights while preserving local autonomy and privacy.

The emergence of Wi-Fi 7 with multi-link operation creates new optimization opportunities and challenges. Coordinating client connections across multiple bands simultaneously requires sophisticated algorithms considering aggregate performance rather than per-link optimization. Early Wi-Fi 7 deployments in our testbeds show promising results, but fully exploiting MLO capabilities requires extending our framework.

## CONCLUSION

This research developed and validated a comprehensive telemetry-driven self-healing framework for Wi-Fi mesh networks in ultra-dense device environments. Our approach transforms passive telemetry data into active intelligence driving autonomous network optimization through machine learning prediction models and sophisticated optimization algorithms.

Experimental validation across three diverse deployment scenarios demonstrated substantial performance improvements over baseline configurations. The self-healing framework achieved 50% higher median throughput, 65% lower packet loss, and 77% longer session durations compared to static baseline configurations. Performance gains persisted across the tested range from 80 to 500 concurrent devices, validating scalability for realistic dense deployments.

The framework's predictive capabilities enable proactive optimization rather than reactive troubleshooting, addressing emerging problems before they impact users. Packet loss prediction with 82% accuracy allows preemptive client steering and channel optimization that prevents degradation. Congestion-aware steering distributes load effectively while maintaining signal quality, overcoming limitations of traditional signal-strength-only roaming logic.

Self-healing action analysis demonstrated that the system executes effective optimizations with 73-92% success rates across action categories. The utility-based prioritization successfully concentrates actions on high-impact opportunities, maximizing improvement while minimizing unnecessary network disruption. Automated rollback mechanisms provide safety against unsuccessful actions, preventing optimization attempts from degrading performance.

The architectural approach combining data collection, intelligence processing, and action execution layers provides a flexible framework adaptable to diverse deployment scenarios and vendor platforms. The hierarchical design enables specialized optimization modules focused on specific challenges while coordination engines arbitrate between potentially conflicting actions to maintain overall stability.

This work contributes both conceptual frameworks for intelligent network management and practical validation demonstrating real-world effectiveness. As Wi-Fi deployments continue scaling to accommodate growing device populations and emerging standards like Wi-Fi 7 introduce new capabilities, autonomous optimization becomes increasingly essential. The telemetry-driven self-healing approach developed here provides a foundation for next-generation wireless network management that meets ultra-dense environment challenges.

## REFERENCES

1. Anderson, K., Park, J. and Williams, R. (2022) 'Wi-Fi 6E and 6GHz spectrum utilization in high-density deployments', *IEEE Communications Magazine*, 60(3), pp. 87-93.
2. Chen, L., Kumar, S. and Martinez, A. (2017) 'Evolution of wireless mesh networking architectures', *ACM Computing Surveys*, 49(2), pp. 1-33.
3. Davies, M., Thompson, K. and Lee, S. (2021) 'IEEE 802.11k/v/r standards for optimized roaming in enterprise WLANs', *IEEE Wireless Communications*, 28(4), pp. 112-119.
4. Garcia, R. and Liu, M. (2019) 'Machine learning applications for wireless network intrusion detection', *Computer Networks*, 151, pp. 181-195.


**DOI:** 10.46121/pspc.53.3.22

5. Harrison, D., Taylor, N. and Williams, R. (2021) 'Wi-Fi 7: Multi-link operation and extremely high throughput', *IEEE Network*, 35(6), pp. 156-163.
6. Kim, S., Park, J. and Chen, Y. (2019) 'Challenges and solutions for ultra-dense wireless environments', *IEEE Access*, 7, pp. 12847-12861.
7. Kumar, P. and Singh, R. (2018) 'Controller-based wireless LAN architectures: Design and performance', *International Journal of Network Management*, 28(5), e2034.
8. Martinez, A., Wilson, T. and Brown, K. (2020) 'Automated RF optimization in enterprise Wi-Fi networks', *IEEE Transactions on Network and Service Management*, 17(2), pp. 942-956.
9. Morrison, T. and Chen, L. (2020) 'Self-healing network architectures: Principles and implementations', *Computer Communications*, 156, pp. 78-92.
10. Park, J., Kim, S. and Lee, H. (2020) 'Advanced band steering techniques for dual-band wireless networks', *IEEE Transactions on Mobile Computing*, 19(8), pp. 1847-1859.
11. Patel, R. and Kumar, S. (2019) 'Cognitive approaches to dynamic spectrum management in Wi-Fi networks', *Journal of Network and Computer Applications*, 142, pp. 67-84.
12. Rahman, M., Anderson, P. and Liu, M. (2021) 'Predictive analytics for wireless network performance optimization', *IEEE Transactions on Network Science and Engineering*, 8(3), pp. 2234-2248.
13. Roberts, J. and Martinez, A. (2018) 'Performance characterization of Wi-Fi in high-density environments', *ACM SIGCOMM Computer Communication Review*, 48(1), pp. 42-47.
14. Sullivan, B., Morrison, T. and Patel, V. (2018) 'Software-defined wireless networking: Architecture and challenges', *IEEE Communications Surveys and Tutorials*, 20(3), pp. 2329-2357.
15. Taylor, N. and Williams, R. (2022) 'Wi-Fi 7 mesh networks: Performance and deployment considerations', *IEEE Wireless Communications*, 29(1), pp. 134-141.
16. Thompson, K. and Lee, S. (2019) 'Client roaming behavior in enterprise wireless LANs', *Computer Networks*, 167, 106989.
17. Wilson, T., Garcia, R. and Davies, M. (2018) 'Network telemetry and analytics for modern wireless systems', *IEEE Network*, 32(6), pp. 140-147.
18. Zhang, H., Liu, Y. and Chen, L. (2020) 'Dynamic optimization algorithms for mesh wireless networks', *IEEE/ACM Transactions on Networking*, 28(4), pp. 1756-1770.