MID-TERM REQUIREMENTS: RRM-PLUS SYSTEM

Overview

Goal: Prove value of the additional radio + client-aware signals for safer, better RRM; ship a pilot feature set.

---

1. ADDITIONAL-RADIO SENSING & SCHEDULING

Requirements

1.1 Adaptive Dwell & Scheduling

- **Objective**: Implement dynamic per-channel dwell times to optimize spectrum scanning
- **Method**: Multi-armed bandit algorithm to focus on likely-noisy channels
- **Constraints**:
  - Must respect DFS (Dynamic Frequency Selection) pre-scan requirements
  - Adapt dwell time based on channel history and interference likelihood
  - Balance exploration (new channels) vs exploitation (known noisy channels)

1.2 Non-Wi-Fi Classifier

- **Objective**: Classify non-Wi-Fi interferers in real-time
- **Implementation**: On-device lightweight CNN/feature-engine
- **Target Interferers**:
  - Bluetooth Low Energy (BLE)
  - Zigbee
  - Microwave ovens
  - FHSS (Frequency Hopping Spread Spectrum) devices
  - Cordless phones
  - Analog video transmitters
- **Output Parameters**:
  - Confidence score (probability of classification)
  - Duty cycle (% time active)
  - Center frequency
  - Bandwidth
- **Performance Requirements**:
  - Low CPU/RAM footprint
  - Real-time classification (sub-second latency)
  - High precision and recall per interference class

1.3 Change Detection

- **Objective**: Detect meaningful spectrum changes rapidly
- **Methods**:
  - Online CUSUM (Cumulative Sum Control Chart)
  - EWMA (Exponentially Weighted Moving Average)
- **Monitored Metrics**:
  - Airtime utilization
  - CCA (Clear Channel Assessment) busy time
  - Noise floor variations
- **Performance Target**: Flag meaningful shifts in seconds, not minutes

1.4 Zero-Impact Serving

- **Objective**: Ensure sensing doesn't degrade client service
- **SLA**: Additional radio sensing must consume <2% of total AP airtime
- **Implementation**:
  - Dedicated radio for sensing (doesn't impact serving radios)
  - Monitor and enforce airtime budget
  - Graceful degradation if budget exceeded

Deliverables

1. **Sensing Pipeline Architecture**
   - Design document with data flow
   - API specifications
   - Integration points with RRM controller
2. **Classifier Metrics Report**
   - Precision/recall per interference class
   - Confusion matrix
   - CPU/RAM budget analysis
   - Latency measurements
3. **API for Downstream RRM**
   - RESTful or gRPC endpoints
   - Data schemas (JSON/Protobuf)
   - Event streaming interface

---

2. CLIENT-VIEW ACQUISITION (STANDARDS FIRST)

Requirements

2.1 802.11k Implementation

- **Objective**: Obtain client-side perspective on RF environment
- **Measurement Types**:

  - **Beacon Report**: Client scans for neighboring APs
  - **Link Measurement**: Request/response for RSSI/SNR measurement
  - **Neighbor Report**: Provide/request list of candidate APs for roaming
- **Data Collected**:
  - Client-side RSSI (Received Signal Strength Indicator)
  - Client-side SNR (Signal-to-Noise Ratio)
  - Neighbor AP rankings from client perspective
- **Implementation Requirements**:
  - Request scheduling to avoid overwhelming clients
  - Handle non-responsive clients gracefully
  - Store historical measurements for trend analysis

2.2 802.11v BSS Transition Management (BSS-TM)

- **Objective**: Intelligent client steering based on combined AP + client view
- **Implementation**:
  - Craft ranked neighbor lists using:
    - Client-view RSSI/SNR data
    - AP load metrics (connected clients, airtime utilization)
    - Channel conditions
  - Send BSS-TM requests to steer clients
- **Metrics to Track**:
  - Client acceptance rate (% clients that honor BSS-TM request)
  - Post-roam QoE deltas (throughput, latency, retry rate changes)
  - Time to complete roaming
  - Failed roam attempts

2.3 Passive Client Inference

- **Objective**: Estimate client perspective without active probing
- **Methods**:
  - **MCS Asymmetry**: Compare uplink vs downlink MCS rates
  - **ACK Timing Variance**: Analyze ACK frame timing patterns
  - **Retry Asymmetry**: Compare uplink vs downlink retry rates (hidden-node proxy)
- **Use Cases**:
  - Detect hidden node problems
  - Identify clients with poor RF conditions
  - Supplement 802.11k data for non-supporting devices

2.4 Optional Synthetic Clients

- **Objective**: Ground-truth validation of RF conditions
- **Implementation**:
  - Deploy USB/SoC Wi-Fi probes at strategic locations

    - 2-5 probes per site recommended
    - Continuously measure:
        - Throughput to/from APs
        - RSSI/SNR at fixed locations
        - Interference patterns
- **Use Cases**:
    - Validate ML model predictions
    - Calibrate passive inference algorithms
    - Baseline for A/B testing

Deliverables

1. **Telemetry Schema**
    - Data models for all measurement types
    - Time-series format specifications
    - Storage schema (time-series DB or data lake)
    - Retention policies
2. **Success Matrix by Device OUI/OS**
    - Breakdown of 802.11k/v support by:
        - Device manufacturer (OUI - Organizationally Unique Identifier)
        - Operating System (iOS, Android, Windows, macOS, Linux)
        - Chipset vendor
    - Success rates for each measurement type
    - Known limitations and workarounds
3. **Privacy Posture Document**
    - Data handling policies
    - PII (Personally Identifiable Information) controls
    - MAC address hashing strategy
    - Data retention windows
    - Opt-out mechanisms
    - GDPR/CCPA compliance considerations
4. **MDM Guidance**
    - Configuration recommendations for Mobile Device Management
    - No agent mandated approach
    - Optional agent benefits if deployed
    - Enterprise deployment best practices

---

3. POLICY & GUARDRAILS (RULES + FIRST ML)

Requirements

3.1 Policy Engine

- **Objective**: Express and enforce Service Level Objectives (SLOs)
- **SLO Examples**:

    - Maximize P50 (median) throughput
    - Keep P95 retry rate < 8%
    - Limit AP config churn   0.2 changes/AP/day
    - Maintain P95 latency < 50ms for voice SSID
    - Ensure minimum RSSI > -70 dBm for 95% of clients
- **Implementation Requirements**:
    - SLO definition language (YAML/JSON)
    - Per-SSID and per-role policy support
    - Conflict resolution when SLOs compete
    - Priority/weight assignment for objectives

3.2 Bayesian Optimizer (BO)

- **Objective**: Tune RRM parameters intelligently
- **Optimized Parameters**:
    - Channel width (20/40/80/160 MHz)
    - Transmit power (TPC - Transmit Power Control)
    - OBSS-PD (Overlapping BSS Preamble Detection) thresholds
- **Optimization Scope**: Per AP-cell basis
- **Constraints**:
    - Regulatory limits (FCC, ETSI, etc.)
    - EIRP (Effective Isotropic Radiated Power) maximums
    - DFS channel restrictions
    - Co-channel and adjacent-channel interference limits
- **Two-Phase Approach**:
    - **Offline Simulation**:
        - Train on historical data
        - Validate on held-out time periods
        - Establish safety baselines
    - **Limited Online BO**:
        - Controlled A/B testing on pilot floors
        - Safety constraints enforced
        - Gradual rollout based on confidence

3.3 Churn Control

- **Objective**: Prevent excessive configuration changes while maintaining agility

- **Mechanisms**:

  Change Budgets:

    - Maximum 1 channel/power/width change per AP per 4-hour window
    - Exception: critical incidents (DFS events, severe interference)
    - Separate budgets for different change types

  Hysteresis:

  - Minimum delta thresholds before change:
    - Power: 2 dB change required
    - Channel width: 1 step change (e.g., 20→40 MHz)
    - Channel: must show sustained improvement
  - Prevent oscillation between similar states

Time-of-Day Windows:

  - Define maintenance windows for non-critical changes
  - Avoid changes during peak hours (unless SLO breach imminent)
  - Respect site-specific schedules (e.g., exam halls)

Cool-Off Periods:

  - Mandatory wait time after any change
  - Extended cool-off after client complaints
  - Progressive backoff for repeated changes
  - Per-AP and per-RF-domain cool-offs

Deliverables

1. **Safe-Change Planner**
   - Architecture and design document
   - Change proposal evaluation logic
   - Safety constraint checker
   - Change scheduling algorithm
   - Blast-radius calculator (limit simultaneous changes)
2. **A/B Toggle System**
   - Feature flag infrastructure
   - Experiment definition framework
   - Control vs treatment group assignment
   - Statistical significance testing
   - Metrics collection and comparison
3. **Rollback Mechanism**
   - Automatic rollback triggers:
     - KPI degradation > X% for Y minutes
     - SLO violations
     - Client complaint threshold exceeded
   - One-click manual rollback interface
   - Configuration versioning and audit trail
   - Rollback testing and validation
4. **First BO Results on Pilot Floor**
   - Experiment setup documentation
   - Baseline measurements
   - Optimization trajectory (parameter changes over time)
   - KPI improvements vs baseline
   - Lessons learned and tuning recommendations

4. KPIs & ACCEPTANCE CRITERIA

Performance Metrics

4.1 QoE (Quality of Experience) Lift

- **Target**: +15-20% median downlink throughput for edge clients
- **Edge Client Definition**: Clients with RSSI between -70 to -65 dBm
- **Measurement**:
  - 60-second measurement windows
  - Per-client median calculation
  - Aggregate across all edge clients
  - Compare baseline vs RRM-Plus performance

4.2 Reliability Improvements

- **Retry Rate Reduction**:
  - Target: P95 retry rate reduced by 20%
  - Calculation: (MAC retries / total MPDUs) at P95
  - Per-SSID reporting
- **Uplink PER Reduction**:
  - Target: P95 uplink PER reduced by 15%
  - Measurement: Per-client P95 across 1-minute bins
  - Focus on clients that were previously problematic

4.3 Steering Efficacy

- **BSS-TM Acceptance Rate**:
  - Target: >85% acceptance on capable clients
  - Measurement: Fraction of BSS-TM responses with target match
  - Segment by device OUI/OS
- **Post-Steer SINR Improvement**:
  - Target: P90 post-steer SINR improvement of +3 dB
  - Measure SINR before and after steering
  - Validate that steers improve client conditions

4.4 Stability Metrics

- **Config Churn**:
  - Target: 0.2 changes/AP/day
  - Measurement: Mean automated changes per AP per day
  - Excludes operator-initiated changes
- **DFS Handling**:
  - Requirement: DFS radar detection events must not impact client service
  - Pre-CAC (Channel Availability Check) on additional radio

    - Seamless channel switching for clients

4.5 Overhead

- **Additional-Radio Sensing Cost**:
    - Target: <2% airtime consumption
    - Calculation: (additional-radio scan airtime / total AP airtime) × 100
    - Continuous monitoring and enforcement

Measurement Requirements

- **Baseline Period**: Minimum 7 days of pre-deployment data
- **Pilot Period**: Minimum 14 days of RRM-Plus operation
- **Comparison Method**: Paired t-test or Mann-Whitney U test for statistical significance
- **Segmentation**: Report all metrics by:
    - SSID/network
    - Device OUI/OS
    - Time of day
    - AP density zone (sparse/medium/dense)

---

5. MID-TERM SUBMISSION REQUIREMENTS

Part 1: Additional-Radio Sensing & Classifier

Components:

1. **Design Document** (15-25 pages)
    - Architecture overview with diagrams
    - Sensing orchestrator design
    - Multi-armed bandit algorithm details
    - CNN/feature-engine architecture
    - CUSUM/EWMA implementation
    - Data flow and processing pipeline
2. **Classifier Results**
    - **Precision/Recall Table**:

Interference Type	Precision	Recall	F1-Score
BLE	0.XX	0.XX	0.XX
Zigbee	0.XX	0.XX	0.XX
Microwave	0.XX	0.XX	0.XX
FHSS	0.XX	0.XX	0.XX
Cordless Phone	0.XX	0.XX	0.XX

    - **Confusion Matrix**: Detailed misclassification analysis
    - **ROC Curves**: Per-class performance curves
    - **Detection Latency**: Time from interference start to classification
3. **Resource Budget Analysis**

  - CPU utilization (% per core)
  - RAM consumption (MB)
  - Power consumption impact
  - Storage requirements
  - Network bandwidth for telemetry
4. **API Documentation**
   - RESTful endpoint specifications
   - Request/response schemas
   - Authentication/authorization
   - Rate limiting
   - Example usage code

Part 2: Safe-Change Planner & Bayesian Optimization

Components:

1. **Safe-Change Planner Documentation** (10-15 pages)
   - Algorithm pseudocode
   - Change evaluation criteria
   - Safety constraint implementation
   - Blast-radius calculation method
   - Change scheduling logic
2. **A/B Testing Framework**
   - Experiment design methodology
   - Control/treatment assignment algorithm
   - Metrics collection infrastructure
   - Statistical testing approach
   - Sample size calculations
3. **Rollback System**
   - Automatic rollback trigger conditions
   - Configuration versioning scheme
   - Audit trail format
   - Manual rollback interface mockups/screenshots
   - Rollback success rate from testing
4. **First BO Results**
   - **Pilot Site Description**:
     - Site type (office, education, retail, etc.)
     - Number of APs
     - Client density
     - Baseline interference profile
   - **Experiment Timeline**:
     - Baseline period dates
     - Optimization period dates
     - Number of iterations/changes
   - **Parameter Evolution**:
     - Channel assignments over time

    - Power settings trajectory
    - OBSS-PD threshold changes
    - Channel width decisions
- **Performance Results**:
    - Before/after KPI comparison table
    - Time-series graphs of key metrics
    - Statistical significance tests
    - Client distribution improvements
- **Lessons Learned**:
    - What worked well
    - Unexpected challenges
    - Tuning recommendations for full deployment

Part 3: Client-View Acquisition & Telemetry

Components:

1. **Telemetry Schema Documentation** (10-15 pages)

    - Data models (JSON/Protobuf schemas)
    - Database schema (if using SQL)
    - Time-series format specifications
    - Data retention policies
    - Query patterns and indexes

2. **802.11k/v Implementation Report**

    - Supported measurement types
    - Request/response handling logic
    - Error handling and retries
    - Client compatibility testing results
    - Performance optimization techniques

3. **Device Support Matrix**

    ```
    OUI/Manufacturer | OS/Version | 802.11k | 802.11v | BSS-TM | Notes
    Apple           | iOS 15+    |   Yes   |   Yes   |  90%   | Excellent support
    Samsung         | Android 12 |   Yes   |   Yes   |  75%   | Some models limited
    ...
    ```

4. **Acceptance Metrics by Device Class**

    - 802.11k response rate by OUI/OS
    - 802.11v BSS-TM acceptance rate by device
    - Passive inference accuracy validation
    - Synthetic client correlation analysis

5. **Privacy & Security Documentation**

    - PII handling procedures
    - MAC address hashing algorithm

- Data retention policy
- Encryption in transit/at rest
- Access control model
- GDPR/CCPA compliance checklist
- Opt-out mechanism description

6. **MDM Deployment Guide**

    - Configuration profiles (iOS, Android)
    - Group policy settings (Windows)
    - Enterprise Wi-Fi setup recommendations
    - No-agent deployment steps
    - Optional agent benefits comparison
    - Troubleshooting guide

Additional Artifacts

1. **Dashboards**
    - Real-time monitoring dashboard screenshots
    - KPI visualization examples
    - Alerting configuration
    - Historical trend views
2. **Offline Replay Simulation**
    - 24-hour log dataset description
    - Simulation framework code
    - Replay methodology documentation
    - Validation results (predicted vs actual)
    - Sensitivity analysis
3. **Pilot Site Report**
    - Executive summary (1-2 pages)
    - Detailed findings (10-15 pages)
    - Before/after comparison
    - ROI analysis
    - Deployment challenges and solutions
    - Recommendations for broader rollout

---

6. TECHNICAL SPECIFICATIONS

6.1 Supported Bands & Channels

- **2.4 GHz**: Channels 1-11 (US), 1-13 (EU)
- **5 GHz**: UNII-1, UNII-2, UNII-2e, UNII-3 bands
- **DFS Channels**: Full support with pre-CAC on additional radio
- **6 GHz**: Architecture extensible (not required for mid-term)

6.2 RRM Levers (Controlled Parameters)

1. **Channel Selection**: Primary channel assignment per AP
2. **Transmit Power**: TPC with 1 dB granularity
3. **Channel Width**: 20/40/80/160 MHz (band-dependent)
4. **OBSS-PD Thresholds**: -82 to -62 dBm range
5. **Target RSSI**: For rate control algorithms
6. **Roam/Steer Hints**: 802.11k/v recommendations
7. **Band Steering**: 2.4 GHz   5 GHz preference
8. **Load Balancing**: Client distribution across APs
9. **Admission Control**: Max clients per AP/SSID

6.3 Monitored Signals

1. **Spectrum Analysis**:
    - FFT (Fast Fourier Transform) data
    - IQ (In-phase/Quadrature) samples
    - Power spectral density
2. **Wi-Fi MAC Counters**:
    - MCS (Modulation and Coding Scheme) rates
    - PER (Packet Error Rate)
    - Retry counts and rates
    - RU (Resource Unit) allocation (11ax)
    - Airtime utilization
3. **Client RF Metrics**:
    - RSSI per client
    - SNR per client
    - 802.11k measurement reports
    - 802.11mc RTT (if supported)
4. **Transport Layer** (optional for mid-term, foundation for end-term):
    - TCP RTT estimates
    - QUIC connection metrics
    - Jitter measurements
    - Application QoS markers

6.4 Guardrail Parameters

1. **Change Budgets**:
    - Per-AP: 1 change per 4 hours
    - Per-RF-domain: N concurrent changes (N = sqrt(total_APs))
    - Global: Max 10% of APs changing simultaneously
2. **Backoff Timers**:
    - Post-change: 15 minutes minimum
    - Post-rollback: 1 hour minimum
    - Post-incident: 30 minutes minimum
3. **Locality Constraints**:
    - Changes to neighbor APs must be staggered by 5 minutes

    - Max 1 channel change per RF neighborhood per hour
4. **Blast-Radius Isolation**:
    - Define RF domains based on AP coupling
    - Limit changes to <20% of domain simultaneously
    - Critical APs (high client count) change last
5. **SLO-Driven Policy**:
    - No changes if SLOs currently met with >10% margin
    - Prioritize changes for SLO-violating APs/SSIDs
    - Emergency mode if P95 metrics exceed 2× threshold

---

7. DEVELOPMENT & TESTING REQUIREMENTS

7.1 Development Environment

- **Languages**: Python 3.9+ (ML/orchestration), C/C++ (on-device classifier)
- **ML Frameworks**: PyTorch or TensorFlow for classifier training
- **Optimization**: Scipy, Scikit-optimize, or GPyOpt for Bayesian Optimization
- **Data Storage**: Time-series DB (InfluxDB, TimescaleDB) or Data Lake (Parquet on S3)
- **API Framework**: FastAPI, Flask, or gRPC
- **Monitoring**: Prometheus + Grafana or equivalent

7.2 Testing Strategy

Unit Testing: - 80%+ code coverage - Classifier performance tests - API endpoint validation - Change planner logic verification

Integration Testing: - End-to-end data pipeline - 802.11k/v request/response flow - BO integration with policy engine - Rollback mechanism validation

Simulation Testing: - Offline replay with historical data - Synthetic interference injection - Multi-site scenario testing - Stress testing with high churn rates

Pilot Testing: - 2 pilot sites minimum (contrasting environments) - 14+ days continuous operation - Daily KPI monitoring - Weekly review meetings - Incident response testing

7.3 Documentation Requirements

- Architecture Decision Records (ADRs)
- API documentation (OpenAPI/Swagger)
- Deployment runbooks
- Troubleshooting guides
- User manuals for operators

- Training materials

---

8. SUCCESS CRITERIA SUMMARY

Must-Have (Mandatory)

Additional radio sensing with <2% airtime impact

Non-Wi-Fi classifier with >70% precision/recall per class

802.11k/v implementation with >50% device coverage

Safe-change planner with automatic rollback

+15% median throughput improvement for edge clients

Config churn 0.2 changes/AP/day

All deliverables from sections 5.1, 5.2, 5.3

Should-Have (Highly Desired)

+20% median throughput improvement

>85% BSS-TM acceptance rate

P95 retry rate reduced by 25%+

Working BO on pilot floor with documented results

Passive client inference with validation

Nice-to-Have (Bonus)

Synthetic client deployment and correlation

>70% device coverage for 802.11k/v

Multiple pilot sites (>2)

7-day offline simulation

Advanced visualization dashboards

---

9. TIMELINE & MILESTONES

Suggested 12-Week Schedule

Weeks 1-2: Foundation - Environment setup - Data collection infrastructure - Initial 802.11k/v implementation - Baseline measurements

Weeks 3-5: Sensing & Classification - Additional radio orchestrator - Non-Wi-Fi classifier development - Change detection algorithms - Unit testing

Weeks 6-8: Policy & Optimization - Safe-change planner - Bayesian optimizer implementation - Churn control mechanisms - Integration testing

Weeks 9-10: Pilot Deployment - Pilot site 1 deployment - Daily monitoring - Issue resolution - Parameter tuning

Weeks 11-12: Analysis & Submission - Data analysis and KPI calculation - Documentation completion - Dashboard creation - Final report and submission