2026 International Conference on Computing, Networking and Communications (ICNC): Next generation of mobile networks (6G and beyond)

NextG Base Station Placement and Failure Mitigation

Shikhar Bhattarai*, Asim Mahat†, Dn Pokhrel*, and Hyuck M. Kwon*
*Department of Electrical and Computer Engineering, Wichita State University, Wichita, KS, USA
†School of Computing, Wichita State University, Wichita, KS, USA
Email: {sxbhattarai13, axmahat1, dxpokhrel}@shockers.wichita.edu, hyuck.kwon@wichita.edu

***Abstract*—The deployment of resilient NextG networks necessitates a dual-focus approach: optimal base stations (BS) placement and efficient operational maintenance. This paper presents an integrated two-stage framework that systematically addresses both objectives. Stage one introduces a GPU-accelerated methodology for BS placement, employing K-Means clustering for initial seeding, followed by a novel hybrid metaheuristic—Fuzzy Role-based Foraging Optimization (FRFO)—to maximize population coverage. Performance evaluation is conducted under the 3GPP Rural Macrocell (RMa) propagation model. FRFO is rigorously benchmarked against six established algorithms (including K-means). Results indicate that FRFO achieves significantly faster convergence and enhanced coverage performance compared with state-of-the-art algorithms. Stage two builds on the optimal BS deployment by introducing a novel, density-aware pathfinding algorithm for dispatching autonomous ground robots to repair failed BSs. This algorithm accounts for both distance and turning delays and is evaluated against standard Dijkstra and A* methods. Simulation results validate that the proposed framework offers a comprehensive solution across the network lifecycle—from initial deployment to maintenance—achieving enhanced coverage and reduced downtime, thereby supporting the robust operation of next-generation wireless networks.**

***Index Terms*—5G, Base Station, Metaheuristic, Failure-mitigation, NextG.**

I. Introduction

In base station (BS) placement problem, we seek the optimal geographic locations for a set of BSs to maximize an objective such as coverage. [1] studies BS placement problems using supervised machine learning. More recently, a related study [2] examined the optimal positioning of edge servers through the Reinforcement Learning (RL) techniques. In [3], the authors proposed similar optimization framework for the efficient deployment of antennas in Distributed Antenna Systems (DAS). Exact BS placement optimization on large-scale instances is computationally prohibitive; hence, a promising feasible solution is the deployment of sophisticated intelligent algorithms—particularly metaheuristic approaches—that can effectively navigate high-dimensional search spaces and yield near-optimal solutions within practical runtimes. For instance, the Whale Optimization Algorithm (WOA) was employed in [4] to address the dynamic sensor deployment problem.

Beyond BS placement, this paper also addresses the challenge of BS failure mitigation. While drones offer rapid response capabilities, their practical deployment is limited by several factors: privacy concerns from hovering near residences, aerodynamic noise, limited battery life, vulnerability to bird strikes, and sensitivity to wind turbulence. Moreover, civil aviation regulations may restrict their use, particularly in populated areas. These issues highlight the value of ground-based robots, which provide a safer, quieter, and more privacy-preserving alternative for restoring network functionality repairing the failed BS reaching the location on time.

Regarding path finding, Dijkstra's and A* are among the most standard algorithms. Recently, many new sophisticated methods including metaheuristics described above, as well as learning-based methods and hybrids have been used for autonomous mobile robot path planning [5].

II. Research Gaps and Contributions

Lack of studies on large-scale rural deployments: Most BS placement studies focus on dense urban areas with small coverage zones. For example, Zhou and Abawajy (2025) used deep RL for 3D Unmanned Aerial Vehicles (UAV) BS placement in a $500 \times 500\,\mathrm{m}^2$ urban area with four UAVs serving 100. Rosslyn city NLOS deployment covered only $488 \times 510\,\mathrm{m}^2$ [6]. Research on large-scale rural deployments—spanning thousands of kilometers, involving real consumer data, and deploying hundreds of base stations—remains scarce. **Gaps in metaheuristics:** Walton *et al.* [7] note that metaheuristics are commonly tested on problems with known optima and clear stopping rules. In contrast, real-world problems lack such clarity, and exact optima may not be meaningful. They argue designers care more about practical improvement within reasonable computational cost than about achieving mathematically optimal solutions. **Lack of a holistic study:** While prior research have individually explored optimal BS placement and route planning—particularly in the context of UAVs—the joint integration of the placement problem with robust ground-based robotic agents remains, to the best of our knowledge, largely unexplored.

The primary **contributions** of this work are summarized as follows:

Proposal of FRFO: A hybrid metaheuristic, Fuzzy Role-based Foraging Optimization (FRFO) algorithm, is proposed for the base station placement problem. FRFO introduces fuzzy movement, memory and taboo list and dynamic role-

This work was supported in part by National Science Foundation (NSF) under Grants 2226447, 2326898, and 2515378.

switching mechanism, to accelerate convergence and to avoid local optimum.

GPU-Accelerated Two-Stage Framework: We present a holistic framework that bridges strategic network design and operational logistics. The first stage leverages **GPU acceleration** to efficiently solve the optimal placement of base stations over a large, real-world dataset, using the complex 3GPP RMa propagation model. The second stage uses these locations as the foundation for a dynamic failure response simulation.

Novel Density-Aware Pathfinding Algorithm: For the operational stage, a new pathfinding algorithm is developed for dispatching autonomous repair robots. The algorithm's weight function uniquely incorporates both road length and population density, allowing it to find more efficient routes by avoiding likely areas of congestion.

Comprehensive Benchmarking and Analysis: The framework's components are rigorously evaluated. In the **GPU-accelerated** placement stage, FRFO is benchmarked against five established metaheuristics and K-means to prove its effectiveness. In the failure-mitigation stage, the proposed pathfinding algorithm is systematically compared against standard Dijkstra and A* algorithms to demonstrate its superior performance in reducing realistic travel time.

III. Problem 1: Placement of BSs

The high-resolution dataset [8] provides latitude and longitude of residents within every 30×30 m^2 area but is organized at the country level. From the U.S. dataset, we extract the GPS boundaries of rural Cimarron County, Oklahoma via Google Maps. This rural region, covering 4752.11 km^2, was selected for BS placements due to its vast area and low population density of 0.4489 /km^2.

Path Loss Model and Fitness Function

The path loss (PL) between a base station and a UE is calculated according to the **Rural Macro (RMa)** scenario defined in the 3GPP technical report TR 38.901 [9]. Our simulation framework implements the complete set of equations for both Line-of-Sight (LOS) and Non-Line-of-Sight (NLOS) conditions as detailed in **Table 7.4.1-1** of that document. This model was chosen for its detailed and realistic characterization of propagation in large-scale rural environments. Simulations were done for carrier frequency of 3.5 GHz. For the average building height (h_b) we utilize a value of 8 m, which is a valid selection within the model's applicability range of 5 m to 50 m, instead of the table's suggested default of 5 m. In addition, shadow fading standard deviation of 6 dB is chosen for all LOS (Line of Sight) conditions. All formulas, including those for LOS probability, as well as parameters such as the base station height (35 m) and user equipment (UE) height (1.5 m), follow the default values defined for the RMa scenario.

The Base Station Placement (BSP) problem is modeled by considering a set of N discrete user locations, $U = \{u_1, u_2, \ldots, u_N\}$, distributed across a defined geographical area. Each user $u_i \in U$ is characterized by coordinates (lat_i, lon_i) and an associated population weight p_i. The objective is to deploy K base stations $S = \{s_1, s_2, \ldots, s_K\}$, where each station $s_j \in S$ has coordinates (lat_j, lon_j). A complete set of BS coordinates $X = \{(lat_j, lon_j) \mid j = 1, \ldots, K\}$ constitutes a candidate solution.

The primary optimization goal is to maximize population coverage. A user u_i is considered covered if the Reference Signal Received Power (RSRP) from its serving BS (the one providing maximum RSRP) is greater than or equal to a threshold $RSRP_{\min}$. The coverage indicator function is

$$C(u_i, X) = \begin{cases} 1, & \text{if } \max_{j=1,\ldots,K} \{RSRP(u_i, s_j)\} \geq RSRP_{\min}, \\ 0, & \text{otherwise.} \end{cases} \tag{1}$$

The overall fitness function $F(X)$, which the metaheuristic maximizes, is the percentage of the total population covered:

$$\text{Maximize} \quad F(X) = \frac{\sum_{i=1}^{N} p_i C(u_i, X)}{\sum_{i=1}^{N} p_i} \times 100\%. \tag{2}$$

A. Proposed FRFO Algorithm

The FRFO algorithm is a hybrid metaheuristic that balances exploitation and exploration by dividing its population into two adaptive roles: **Workers** and **Scouts**. Workers refine known solutions, while Scouts search for new possibilities. Agents dynamically switch roles based on their performance, guided by a stagnation-based mechanism.

1) Movement Equations: Worker movement is based on a modified PSO strategy, while Scouts use a Lévy flight for exploration.

Worker (Exploitation):

$$\begin{aligned} V_i(t+1) = & wV_i(t) + c_1 r_1 (P_{best,i}(t) - X_i(t)) \\ & + c_2 r_2 g_{\text{factor}} (G_{best}(t) - X_i(t)) \end{aligned} \tag{3}$$

$$X_i(t+1) = X_i(t) + V_i(t+1) \tag{4}$$

$$g_{\text{factor}} = 0.5 + \frac{\|G_{best} - X_i\|}{\|X_{max} - X_{min}\| + \epsilon} \tag{5}$$

Scout (Exploration):

$$X_i(t+1) = X_i(t) + \alpha \oplus \text{Levy}(\beta) \tag{6}$$

where X_i is the position and V_i is the velocity of agent i at iteration t. $P_{best,i}$ is the agent's personal best position, and G_{best} is the global best. Coefficients include inertia weight w, acceleration constants c_1, c_2, and random numbers $r_1, r_2 \in [0, 1]$. The term g_{factor} is a fuzzy component that modulates social influence based on the agent's distance ($\|\cdot\|$) from G_{best} within the search space boundaries (X_{max}, X_{min}). Fuzzy movement dynamically adjusts an agent's attraction to the best global solution by strengthening the pull when the agent is far away and weakening it to allow for finer, more precise searching when it is close by. ϵ is a small constant to prevent division by zero. For scouts, α is a step size, $\oplus$ is entry-wise multiplication, and Lévy(β) is a random step from a Lévy distribution with index β.

2) Adaptive Control: An agent's role is not fixed. A Worker that fails to improve its solution after a set number of iterations (stagnation) becomes a Scout. Conversely, a Scout that finds a superior solution becomes a Worker to exploit the new region. Unlike conventional algorithms such as the Artificial Bee Colony (ABC), where the transition from worker to scout typically occurs unidirectionally in response to resource stagnation, the FRFO algorithm enables bidirectional role transitions, allowing agents to dynamically switch between worker and scout roles based on environmental cues and performance metrics. To prevent cycling, the algorithm uses a long-term taboo list for stagnant solutions and a short-term memory archive to preserve elite solutions found during the search.

Let T, P, U, and K denote the number of iterations, population size, UE locations, and base stations, respectively. FRFO, ABC, GWO, PSO, and WOA share the same computational complexity $\mathcal{O}(TPUK)$, driven by iterative fitness evaluations over U users and K sites, whereas RPO is roughly twice as expensive due to its dual-phase updates ($2TP$ vs. TP), and the one-time K-Means initialization adds only negligible overhead relative to the metaheuristic search as shown in Fig. 1.

Competitor Algorithms, Parameters and Coefficients: All algorithms share common parameters of a population size (N_{pop}) of 42 and 30 iterations (N_{iter}). For each BS K, the experiment is run 5 times and average is taken. Thus, 150 iterations were run per K. For Particle Swarm Optimization (PSO) and the proposed FRFO algorithm, the inertia weight (w) is 0.7, and cognitive (c_1) and social (c_2) coefficients are 1.5. Both Grey Wolf Optimizer (GWO) and Whale Optimization Algorithm (WOA) use a linearly decreasing parameter (a) from 2 to 0, with WOA adding a logarithmic spiral constant (b) of 1. The Artificial Bee Colony (ABC) sets the number of food sources to half the population size (21) and defines an abandonment limit (L) dependent on the number of clusters (K). Red Panda Optimizer (RPO) operates using the common parameters and internally generated random variables without other specific fixed coefficients. FRFO includes many unique parameters: a Lévy exponent (β_{Levy}) of 1.5, a season length of 10 (good solutions are stored in memory after 10 iterations), a stagnation threshold of 15, a taboo radius of 0.001 (in degrees), a taboo duration of 20, and a memory archive size of 5.

Algorithm 1 Fuzzy Role-based Foraging Optimization (FRFO)

Require: $N, K, T_{\max}$ ▷ Population size, dimensions, max iterations
S_{kmeans} ▷ Initial seed solution from K-Means B ▷ Search space boundaries FRFO parameters: $L_{\text{season}}, \theta_{\text{stag}}, R_{\text{taboo}}, D_{\text{taboo}}$
Ensure: $\mathbf{g}_{\text{best}}, f(\mathbf{g}_{\text{best}})$ ▷ Best solution and its fitness

Initialization:
$\mathbf{P} \leftarrow$ InitializePopulation(N, S_{kmeans}) ▷ Create population around seed
$\mathbf{V} \leftarrow \mathbf{0}$; $\mathbf{P}_{\text{best}} \leftarrow \mathbf{P}$; $f(\mathbf{P}_{\text{best}}) \leftarrow$ EvaluateAll$(\mathbf{P})$
$(\mathbf{g}_{\text{best}}, f(\mathbf{g}_{\text{best}})) \leftarrow$ FindBest$(\mathbf{P}_{\text{best}})$
$\mathbf{C}_{\text{stag}} \leftarrow \mathbf{0}$; $L_{\text{taboo}} \leftarrow \emptyset$; $M_{\text{archive}} \leftarrow \emptyset$ ▷ Init counters and lists
Assign Roles: Randomly assign half of $\mathbf{P}$ as *Workers*, half as *Scouts*

for $t = 1$ **to** $T_{\max}$ **do**
  //— Memory and Taboo List Maintenance —//
  if $t \bmod L_{\text{season}} == 0$ **then**
    $M_{\text{archive}} \leftarrow$ UpdateMemory$(\mathbf{g}_{\text{best}})$
  Prune expired entries from L_{taboo} based on D_{taboo}

  //— Agent Movement Phase —//
  for $i = 1$ **to** N **do**
    if agent i is *Scout* **then**
      $\mathbf{P}_i \leftarrow$ UpdateByLevyFlight$(\mathbf{P}_i)$ ▷ Exploration
    else if agent i is *Worker* **then**
      $\mathbf{V}_i \leftarrow$ UpdateVelocityPSO$(\mathbf{V}_i, \mathbf{P}_i, \mathbf{P}_{\text{best}}[i], \mathbf{g}_{\text{best}})$
      $\mathbf{P}_i \leftarrow \mathbf{P}_i + \mathbf{V}_i$ ▷ Exploitation
    Clip $\mathbf{P}_i$ to boundaries B
    if IsNearTabooLocation$(\mathbf{P}_i, L_{\text{taboo}}, R_{\text{taboo}})$ **then**
      $\mathbf{P}_i \leftarrow$ RelocateFrom$(\mathbf{P}_i, M_{\text{archive}})$ ▷ Jump away from taboo region
      $\mathbf{C}_{\text{stag}}[i] \leftarrow 0$

  //— Evaluation and Fitness Update —//
  for $i = 1$ **to** N **do**
    if $f(\mathbf{P}_i) > f(\mathbf{P}_{\text{best}}[i])$ **then**
      Update $\mathbf{P}_{\text{best}}[i]$ with $\mathbf{P}_i$; $\mathbf{C}_{\text{stag}}[i] \leftarrow 0$ ▷ New personal best
      if $f(\mathbf{P}_{\text{best}}[i]) > f(\mathbf{g}_{\text{best}})$ **then**
        Update $\mathbf{g}_{\text{best}}$ with $\mathbf{P}_{\text{best}}[i]$ ▷ New global best
    else
      $\mathbf{C}_{\text{stag}}[i] \leftarrow \mathbf{C}_{\text{stag}}[i] + 1$ ▷ Increment stagnation counter

  //— Adaptive Role Switching —//
  for $i = 1$ **to** N **do**
    if agent i is a *Worker* **and** $\mathbf{C}_{\text{stag}}[i] > \theta_{\text{stag}}$ **then**
      Switch agent i to *Scout*; $\mathbf{C}_{\text{stag}}[i] \leftarrow 0$
      Add $\{\mathbf{P}_{\text{best}}[i], t + D_{\text{taboo}}\}$ to L_{taboo}
    else if agent i is a *Scout* **and** just found a new personal best **then**
      Switch agent i to *Worker*; $\mathbf{C}_{\text{stag}}[i] \leftarrow 0$

return $\mathbf{g}_{\text{best}}, f(\mathbf{g}_{\text{best}})$

B. Experiment and Results

We take 5G system parameters to verify the proposed methods. However, the scope of this current paper is applicable for the NextG as well. In **Layer 1**, standard K-means clustering is applied to the (lat, lon) coordinate pairs . The centroids of the K clusters serve as the initial BS locations, denoted as X_0. In **Layer 2**, these seeded locations are refined by metaheuristic algorithms, including FRFO, to maximize population coverage. A minimum RSRP coverage threshold of $-90\,\text{dBm}$ and an interferer BS activity factor of 0.8 are used for performance calculations. Additionally, a thermal noise density of -174 dBm/Hz is used, representing the baseline noise power per Hz at room temperature. The bandwidth of each 5G Physical Resource Block (PRB) is taken as 180 kHz, corresponding to 12 subcarriers each spaced at 15 kHz. A UE noise figure of 9 dB is assumed, as recommended by the 3GPP standard for simulations. Using these values, the total noise power is calculated as -112.45 dBm.

As shown in Fig. 2, FRFO consistently outperforms all algorithms in the range $K = 3$ to 8, achieving coverage exceeding 45%. Beyond this range, while Fig. 2 shows FRFO maintaining strong performance, it slightly trails PSO from $K = 9$ onwards, yet remains competitive with all other state-of-the-art methods. FRFO outperforms competing methods for $K = 3 - 8$ because its role-switching and Lévy flights,

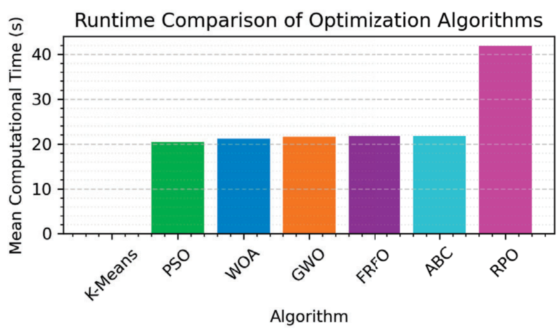

Fig. 1. Runtime comparison

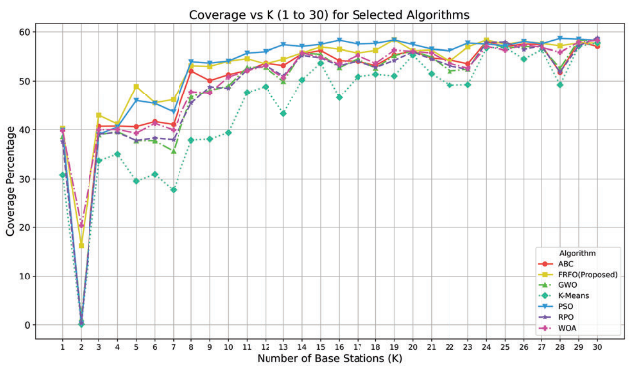

Fig. 2. Coverage up to 30 BS deployments

guided by stagnation detection and a Taboo list, drive agents out of initialization-dependent local optima and across gaps between population clusters, revealing better global solutions than the gradient-like updates in PSO and GWO. A similar trend is observed in Fig. 3, where FRFO continues to lead up to moderate K values, albeit with a smaller margin, and remains closely competitive with PSO.

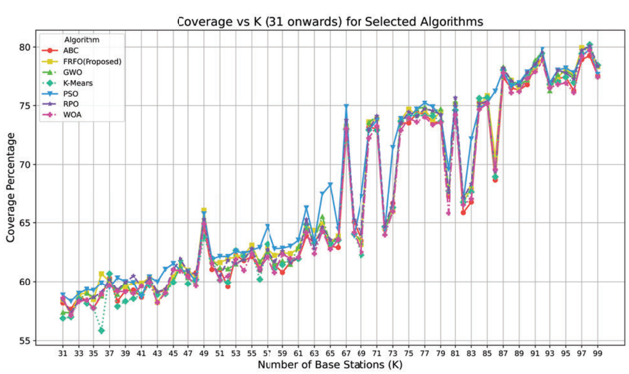

Fig. 3. Coverage for K = 31 BSs and higher

Since the coverage of each metaheuristic depends on initial seed of K-means, the coverage rises or dips as per the K-means initial coverage making the comparison fair. As the number of base stations increases, each UE begins to receive signals from multiple BSs. However, only the strongest signal is used, while the others contribute to interference. This leads to a reduction in the Signal-to-Interference-plus-Noise Ratio

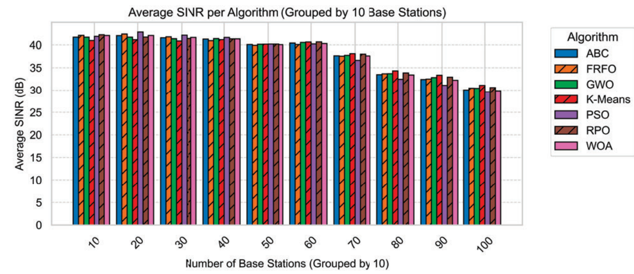

Fig. 4. SINR

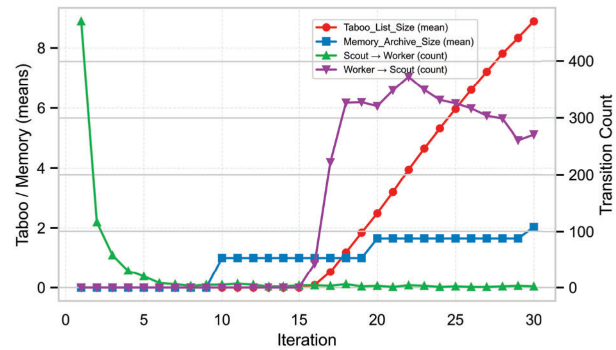

Fig. 5. Memory, taboo and role transitions in FRFO

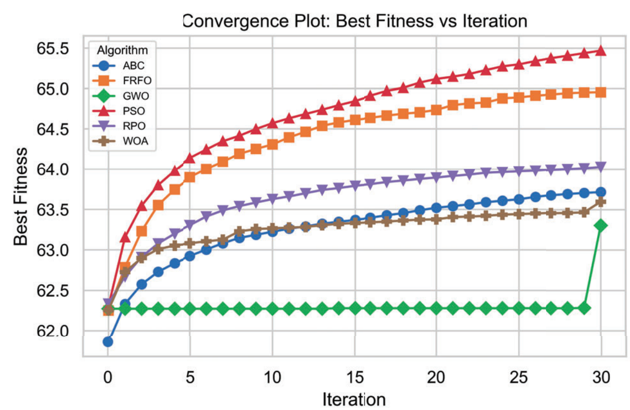

Fig. 6. Convergence as per iterations

(SINR), for higher K as illustrated in Fig. 4. As denser population patches are discovered, a Scout's personal best improves, prompting a role transition to Worker for local exploitation. Conversely, if stagnation occurs, a Worker is demoted to Scout to resume exploration. The right vertical axis of Fig. 5 captures these transitions: the decaying green line indicates frequent Scout-to-Worker shifts in early iterations, while the purple line rises after the stagnation threshold of 15 iterations, reflecting increased Worker-to-Scout transitions. The left vertical axis illustrates how FRFO leverages a short-term *taboo_list* and a *memory_archive*. When an agent's stagnation counter exceeds the threshold, its last position is stored in the taboo list. During the *Taboo and Memory Check*, if an agent revisits a tabooed region, it is relocated—either based on elite solutions from the archive or reinitialized randomly if the archive is empty. This avoids redundant search in recently unproductive areas. Taboo entries expire after 20 iterations, enabling future re-exploration. Memory and transition features along with the fuzzy movement described by Eq. 5, help FRFO find better solutions and hence converge better. The convergence is shown in Fig. 6.

IV. Problem 2: Failure of base stations

To address base station (BS) failures and minimize service downtime, we deploy five standby ground-based robots, each pre-trained to autonomously repair failed BSs. As observed in the coverage plot of Fig. 3, the increase in coverage becomes marginal with increasing number of BSs (K), in contrast to the trend in Fig. 2. This behavior stems from the fact that, beyond a certain point, additional BSs are placed in sparsely populated regions, as the majority of densely populated areas have already been served. Consequently, a typical mobile network operator would consider the diminishing returns of further deployments through a cost-benefit analysis.

Specifically, in Fig. 3, at $K = 67$, approximately 75% of the population is already covered, immediately after which the coverage drops. We select this case ($K = 67$) for our analysis and adopt the BS locations provided by the algorithm that yielded the highest coverage at this value of K, namely Particle Swarm Optimization (PSO).

To minimize both travel time and path distance, the robots are initially positioned using a K-Means clustering approach with $K = 5$. The 67 BS locations obtained from PSO serve as input data points for this clustering. The resulting five cluster centroids represent the initial robot positions, as illustrated by the circular regions labeled “Robot Start” in Fig. 7. Initially, each robot is thus strategically placed at the centroid of a cluster to optimally cover and respond to potential BS failures so that the total time and distance traversed is minimum.

Proposed Method: To enable the robots to traverse from their initial locations to the failed base station (BS) sites, we propose a novel probabilistic routing approach as defined in Equations 7–9. Unlike the classical Viterbi algorithm, where both transition and emission probabilities are explicitly known, such information is not directly available in road network scenarios. Therefore, we introduce a cost-based probability model to govern the robot’s path selection.

In this formulation, each possible path is assigned a cost, and the likelihood of a robot choosing a particular path is inversely related to this cost. Specifically, the cost function in Equation 7 incorporates a penalty for longer distances as well as for routes passing through regions of higher population density. As a result, paths that are shorter and traverse less densely populated areas are probabilistically favored, thus enabling an efficient and context-aware navigation strategy.

While traversing the road network, a robot must consider not only the shortest physical path from its initial location to the failed BS, but also the population density along that route. High-density areas are more likely to exhibit behaviors that introduce additional delay—such as frequent pedestrian crossings, random road crossings, or vehicles entering and exiting residential or commercial zones. Furthermore, regions with dense populations often enforce lower speed limits, which further slow down traversal. These factors are captured by the density term in Equation 7, which penalizes routes passing through densely populated areas.

In addition to population-induced delays, the presence of traffic signals further impacts the robot’s travel time. To model this, a fixed delay of 30 seconds is incorporated whenever a traffic signal is encountered along the path. This composite formulation of cost—accounting for both physical distance and contextual factors such as population density and traffic signals—constitutes what we term a *hybrid probabilistic model*.

The road map utilized in this study, shown in Fig. 7, is derived from OpenStreetMap data. The overlaid heatmap represents the population density in units of persons per square kilometer, effectively visualizing the environmental constraints that influence the robot’s routing decisions.

Cost and Path Computation

1. Cost Function:

$$\text{Cost}_{uv} = \text{Length}_{uv} + \text{Density}_{v} + \text{Traffic}_{uv} \tag{7}$$

where Length_{uv} is the distance (in meters) between nodes u and v, Density_{v} denotes the local population density (in people per km^2) at node v, and Traffic_{uv} represents a fixed wait time (30 seconds) due to traffic signals, if present along edge (u, v).

2. Transition Probability:

$$P_{uv} = \frac{\frac{1}{\text{Cost}_{uv}}}{\sum_{v' \in \mathcal{N}(u)} \frac{1}{\text{Cost}_{uv'}}} \tag{8}$$

where $\mathcal{N}(u)$ denotes the set of neighboring nodes of node u. Lower-cost paths yield higher transition probabilities.

3. Proposed Path Cost:

$$\text{PathCost}_{uv}^{\text{Proposed}} = 1 - P_{uv} \tag{9}$$

4. Dijkstra and A* Path Cost:

$$\text{PathCost}_{uv}^{\text{Dijkstra}} = \text{PathCost}_{uv}^{\text{A*}} = \text{Length}_{uv} \tag{10}$$

The per-path computational complexity for the Proposed, Dijkstra, and A* algorithms are $\mathcal{O}(N+E)+\mathcal{O}(E+N\log N)$, $\mathcal{O}(E+N\log N)$, and $\mathcal{O}(E+N\log N)$, respectively. Here, N and E denote the number of nodes (intersections) and edges (road segments), and the additional $\mathcal{O}(N+E)$ term in the proposed method is negligible compared to the dominant term, contributing minimal overhead.

A. Comparison on a map

In Fig. 7, the initial positions of the robots are labeled as *Robot 1*, *Robot 2*, and so on. From the set of 67 BSs obtained via the PSO algorithm, we randomly deactivate between 1 and 5 BSs and repeat the experiment over 50

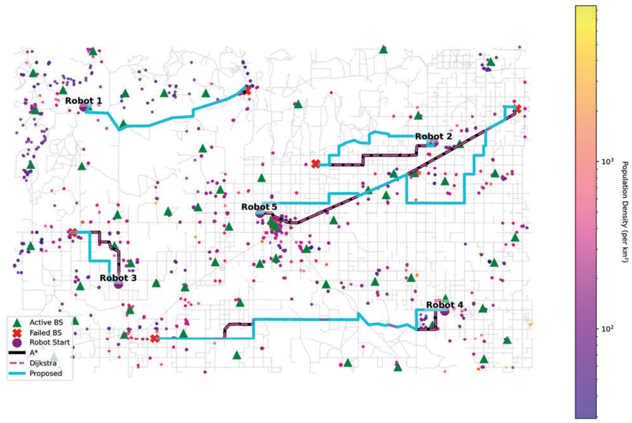

Fig. 7. Path taken by robots upon BS failure in Cimarron County, Oklahoma

independent runs. Fig. 7 illustrates the outcome of one such run where 5 BSs were deactivated. The paths followed by all five robots are plotted using three different path planning strategies: the proposed hybrid probabilistic method, Dijkstra's shortest path algorithm, and A*. As both Dijkstra and A* rely solely on shortest distance, they yield identical paths. In contrast, the proposed method incorporates environmental context—particularly population density—by penalizing routes that pass through densely populated areas. The resulting paths clearly show divergence between the proposed method and the conventional algorithms. Notably, except for *Robot 1*, all other robots encounter high-density regions and adjust their paths to avoid such areas, highlighting the adaptive behavior introduced by the proposed model.

B. Comparison of distance and turns

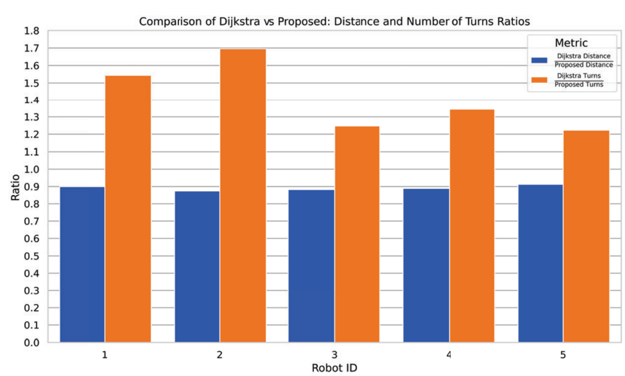

Fig. 8. Comparison of distance and turns

Since Dijkstra and A* follow nearly identical shortest-distance paths, we compare their performance to the proposed method via the Dijkstra-to-Proposed distance ratio in Fig. 8, which stays close to 0.9 for all five robots, indicating only a modest increase in path length. More importantly, the Dijkstra-to-Proposed turn ratio is significantly larger than 1, showing that the proposed method yields substantially fewer turns by avoiding densely populated minor roads with many junctions, where robots must decelerate, face congestion, and are subject to lower speed limits. Consequently, despite slightly longer routes, the proposed method is better suited for emergency failure-mitigation scenarios, where minimizing travel time rather than geometric distance is paramount.

V. Discussion, Conclusion and Future Work

In this paper, we first presented a comparative analysis of base station placement in rural regions, demonstrating the superior performance of the proposed hybrid FRFO algorithm over state-of-the art metaheuristics. Second, we proposed a simple yet effective density-aware path-finding algorithm for robot-based failure mitigation, which, unlike Dijkstra's and A*'s shortest path algorithms, results in trajectories with fewer turns. Although the proposed density-aware method is applied here for navigating robots to failed BSs, its applications are far-reaching. For instance, mobile robots operating as on-demand vending machines could deliver goods more rapidly to customer homes; firefighting robots could be deployed swiftly in response to nearby smoke alarms; and bomb squad robots could reach hazardous sites more quickly in critical scenarios. While the BS placement strategy described herein can lead to significant resource savings for mobile operators, our density-aware robotic path finder holds the potential to save lives. As a future extension of this work, we plan to deploy and evaluate the both proposed methods in a variety of suburban and urban contexts using Uma and Umi path loss models.

References

[1] W. R. Loh, S. Y. Lim, I. F. M. Rafie, J. S. Ho, and K. S. Tze, "Intelligent base station placement in urban areas with machine learning," *IEEE Antennas and Wireless Propagation Letters*, vol. 22, no. 9, pp. 2220–2224, 2023.

[2] Z. Zhou and J. Abawajy, "Reinforcement learning-based edge server placement in the intelligent internet of vehicles environment," *IEEE Transactions on Intelligent Transportation Systems*, pp. 1–11, 2025.

[3] Z. Chen, Y. Liu, G. Sun, X. Zhou, B. Li, S. Liang, and Q. Zhou, "Planning optimization of the distributed antenna system in high-speed railway communication network based on improved cuckoo search," *International Journal of Antennas and Propagation*, vol. 2018, no. 1, p. 3641286, 2018. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1155/2018/3641286

[4] R. Özdağ and M. Canayaz, "A new dynamic deployment approach based on whale optimization algorithm in the optimization of coverage rates of wireless sensor networks," *European Journal of Technique (EJT)*, vol. 7, no. 2, p. 119–130, 2017.

[5] M. Badamasi Aremu, I. K. Kabir, G. Ahmed, and S. El-Ferik, "Autonomous mobile robot path planning techniques—a review: Classical and heuristic techniques," *IEEE Access*, vol. 13, pp. 117 999–118 022, 2025.

[6] S. K. Vankayala, P. L. Mehta, P. N, S. Yoon, K. Sharma, S. Ojha, and S. K. H. U, "Map-assisted outdoor localization of wall-mounted mmwave base stations for nlos coverage," in *2025 IEEE 22nd Consumer Communications Networking Conference (CCNC)*, Jan 2025, pp. 1–2.

[7] S. Walton, O. Hassan, K. Morgan, and M. Brown, "Modified cuckoo search: A new gradient free optimisation algorithm," *Chaos, Solitons Fractals*, vol. 44, no. 9, pp. 710–718, 2011. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S096007791100107X

[8] Humanitarian Data Exchange, "United States - High Resolution Population Density Maps and Demographic Estimates," Available: https://data.humdata.org/dataset, 2020, accessed: August 1, 2025.

[9] "Study on channel model for frequencies from 0.5 to 100 GHz," 3GPP, Technical Report (TR) 38.901, 2024, Release 18. [Online]. Available: https://www.3gpp.org/ftp/Specs/archive/38_series/38.901/38901-1800.zip