This article has been accepted for publication in IEEE Open Journal of the Communications Society. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/OJCOMS.2025.3531318

IEEE Open Journal of the Communications Society

Received 04 November, 2024; revised 22 December, 2024; accepted 15 January, 2025; Date of publication XX Month, XXXX; date of current version 15 January, 2024.

Digital Object Identifier 10.1109/OJCOMS.2024.011100

Harnessing Meta-Reinforcement Learning for Enhanced Tracking in Geofencing Systems

Alireza Famili[1] *(Member, IEEE)*, Shihua Sun [2], Tolga Atalay [2] *(Member, IEEE)*, AND Angelos Stavrou [1,2] *(Senior Member, IEEE)*

[1]WayWave Inc, Arlington, VA 22203 USA
[2]Department of Electrical & Computer Engineering, Virginia Tech, Arlington, VA 22203 USA

CORRESPONDING AUTHOR: Alireza Famili (e-mail: afamili@waywave.com).

ABSTRACT Geofencing technologies have become pivotal in creating virtual boundaries for both real and virtual environments, offering a secure means to control and monitor designated areas. They are now considered essential tools for defining and controlling boundaries across various applications, from aviation safety in drone management to access control within mixed reality platforms like the metaverse. Effective geofencing relies heavily on precise tracking capabilities, a critical component for maintaining the integrity and functionality of these systems. Leveraging the advantages of 5G technology, including its large bandwidth and extensive accessibility, presents a promising solution to enhance geofencing performance. In this paper, we introduce ***MetaFence**: **Meta**-Reinforcement Learning for Geo**fenc**ing **E**nhancement*, a novel approach for precise geofencing utilizing indoor 5G small cells, termed "5G Points", which are optimally deployed using a meta-reinforcement learning (meta-RL) framework. Our proposed meta-RL method addresses the NP-hard problem of determining an optimal placement of 5G Points to minimize spatial geometry-induced errors. Moreover, the meta-training approach enables the learned policy to quickly adapt to diverse new environments. We devised a comprehensive test campaign to evaluate the performance of *MetaFence*. Our results demonstrate that this strategic placement significantly improves tracking accuracy compared to traditional methods. Furthermore, we show that the meta-training strategy enables the learned policy to generalize effectively and perform efficiently when faced with new environments.

INDEX TERMS geofencing, tracking, meta-RL, sensor placement, 5G networks

I. Introduction

The advent of mobile user technologies, such as unmanned aerial vehicles (UAVs), commonly referred to as drones, alongside the burgeoning mixed reality metaverse platforms, introduces significant new security vulnerabilities [1]–[3]. These innovations possess the potential to infiltrate and compromise restricted zones, thereby escalating the risk of security breaches and the exposure of confidential information [4]–[6]. For example, drones are increasingly deployed for unauthorized surveillance of protected areas, including military facilities, presenting substantial security challenges [7]–[9]. Similarly, users within mixed reality environments, such as the metaverse, may manipulate security flaws to illicitly enter private virtual spaces, where they might misappropriate sensitive information or inflict digital disruption [10]–[12].

In response to these emerging security challenges, the implementation of geofencing systems presents a viable solution to enforce virtual boundaries and manage access to sensitive areas [13]. Through real-time monitoring and access controls, geofencing ensures robust regulation of entry points, thereby thwarting unauthorized access. This technology provides a distinct advantage over traditional vision-based surveillance systems like cameras, as it relies on the transmission and reception of radio frequency (RF) signals. Unlike camera systems, geofencing operates independently of lighting conditions, ensuring consistent functionality across diverse environments. Moreover, geofencing systems are typically less costly to implement and maintain than camera-based monitoring, offering a more economical yet effective security solution [14], [15].

Choosing 5G technology, renowned for its high-frequency RF signals, represents an appropriate strategy for enhancing

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License. For more information, see https://creativecommons.org/licenses/by-nc-nd/4.0/

This article has been accepted for publication in IEEE Open Journal of the Communications Society. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/OJCOMS.2025.3531318

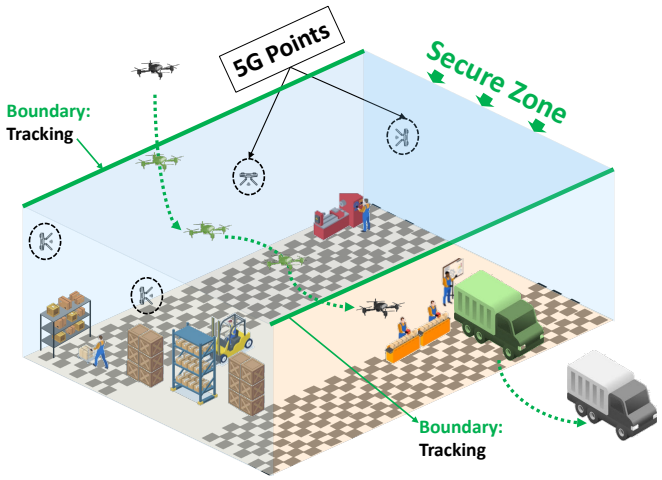


Figure 1: Graphical depiction of the 5G Points deployment within a secure facility, aimed at safeguarding against and tracking unauthorized users.

geofencing capabilities. A significant advantage of 5G is its widespread accessibility, which enables a majority of users to utilize this technology for communication without necessitating additional infrastructure [16]. Essentially, the geofencing system integrates seamlessly with existing 5G base stations (BSs), with all users already equipped to interface with this technology. Furthermore, the ultra-low latency communication combined with expansive bandwidth, which exceeds $400\ MHz$, facilitates a high-resolution geofencing system and ensures high tracking accuracy. In contrast to the foundational IEEE 802.11 standard, commonly known as Wireless Fidelity (Wi-Fi), 5G technology offers superior scalability, including outdoor environments. In comparison, Wi-Fi is primarily designed to support indoor wireless networks, limiting its applicability for extensive geofencing implementations.

Selecting the right wireless technology is crucial, as it directly impacts tracking accuracy through precise ranging measurements such as time of arrival (TOA) and angle of arrival (AOA). These measurements benefit from expansive bandwidth and robust orthogonal frequency-division multiplexing (OFDM) modulation offered in 5G technology, which helps to reduce noise and multipath distortions. Additionally, the spatial arrangement of the BSs significantly influences the accuracy of geofencing systems. The accuracy is determined not only by ranging errors but also by errors caused by the spatial geometric configurations of the BSs.

For outdoor environments with fixed BSs, control over placement to minimize geometric-induced errors is limited. However, in confined venues, geofencing systems offer more flexibility in positioning RF sensors strategically. The compact antenna design due to the high-frequency capabilities of 5G technology facilitates the use of small cells specifically designed for indoor environments. These cells, significantly smaller than their outdoor counterparts, provide comparable cellular coverage using the same core network.

In this paper, our objective is to minimize geometry-induced errors in tracking. To that end, we propose ***MetaFence**: **Meta**-Reinforcement Learning for Geo**fenc**ing **E**nhancement*, a novel framework designed to enhance geofencing capabilities. *MetaFence* utilizes the small indoor cells, which we refer to as "5G Points" as the geofencing anchors. By leveraging an innovative meta-reinforcement learning (meta-RL) methodology, *MetaFence* significantly improves tracking accuracy. This is achieved through the strategic configuration of 5G Points, optimizing their deployment to enhance the effectiveness of geofencing operations.

Our work focuses on exploring the impact of spatial geometry between users and 5G Points. We aim to develop optimal configurations to enhance tracking precision. To rigorously quantify the effect of relative geometry between the user and the 5G Points, we compute the Cramér-Rao lower bound (CRLB) for our tracking estimator. This calculation demonstrates that the error in three-dimensional (3D) tracking using round-trip time (RTT) trilateration arises not only from ranging inaccuracies but also from the spatial positioning of the user relative to the tracking nodes.

The variance in the final 3D tracking error is influenced by the compounded effects of ranging errors and a geometry-induced factor. For example, if the distance ranging error is reduced to 10 cm, a feat achievable with 5G mm-Wave technology, and the geometric-induced error is high—say a factor of 100 due to traditional placements—the total tracking error can escalate to approximately 10 m, calculated as 10 cm $\times$ 100 $=$ 1000 cm. This increase in error, from 10 cm in the distance ranging to 10 m in 3D tracking, is unsuitable for geofencing applications that require high precision.

We first introduce an optimization formulation to tackle the NP-hard problem of determining optimal 5G Point configurations. Following this, we implement a novel meta-RL strategy to identify the most effective deployments for the 5G Points. Previous studies (e.g., [13], [17], [18]), have primarily focused on determining the required number of anchors for a precise geofencing system. However, they fall short in addressing how to optimally configure these anchors once their number is established. Our work fills this gap by focusing on the crucial yet previously overlooked element of optimal anchor configuration.

Traditional Reinforcement Learning (RL) approaches [19] generally tackle each task independently from scratch, resulting in considerable training time overhead when applied to any new tasks. Given the practical need for deploying 5G Points in diverse environments, a more efficient approach is required—one that can quickly adapt to new environments without the need for an extensive and time-consuming training process. To this end, we adopt the meta-RL method [20], [21] as a practical solution for the deployment of 5G Points across various environments. Instead of optimizing each task in isolation, meta-RL trains on a diverse set of tasks to learn a meta-model. This meta-model serves as a robust

This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License. For more information, see https://creativecommons.org/licenses/by-nc-nd/4.0/

This article has been accepted for publication in IEEE Open Journal of the Communications Society. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/OJCOMS.2025.3531318

initialization policy that can be rapidly adapted to new, unseen tasks, specifically finding optimal positions for 5G Point deployment in new environments as discussed in this paper.

While deriving the CRLB for *MetaFence* tracking estimator, we found that the spatial arrangement of 5G Points influences both horizontal and vertical tracking accuracies. Our analyses using coverage heatmaps reveal that vertical accuracy (i.e., height estimation) is more susceptible to the spatial arrangement of the anchors compared to horizontal tracking accuracy (i.e., $X-Y$ plane estimation). Our key contribution is the development of a novel meta-RL-based approach that determines the deployment of 5G Points to reduce spatial geometry-induced errors, simultaneously for vertical and horizontal tracking, enhancing overall geofencing effectiveness. This solution is crucial for applications requiring precise height estimation, such as drone detection, or for mixed reality experiences focused on the $X-Y$ plane tracking. Figure 1 provides a visual representation of *MetaFence* system, and a detailed summary of our contributions is outlined below:

- We present *MetaFence*, an advanced geofencing system that leverages 5G Points for precise user tracking within designated confined venues.
- We calculate the CRLB for the tracking framework of *MetaFence*, attributing the overall 3D tracking error to errors based on range and spatial geometry.
- We formulate the NP-hard problem of optimally deploying 5G Points for enhanced tracking. Our analysis reveals that horizontal tracking is less affected by the spatial geometry of the anchors compared to vertical tracking accuracy.
- We introduce a new strategy utilizing a meta-RL approach to effectively and efficiently tackle the NP-hard challenge of placing 5G Points in diverse indoor environments.
- We conducted an extensive testing campaign to assess the performance of *MetaFence*, comparing our meta-RL-based approach with traditional manual placement strategies. Our results indicate that *MetaFence* significantly reduces spatial geometry-induced errors, thereby markedly enhancing tracking accuracy within the system.

II. Related Works & Background

Our work intersects with three primary research streams: (i) geofencing methodologies, (ii) optimal placement of anchor nodes, and (iii) learning-based problem-solving techniques. Below, we provide a succinct overview of each area.

Geofencing and Tracking: Geofencing is a location-based technology that uses RF signals (e.g., Wi-Fi) to create virtual boundaries around predefined areas, triggering automated actions as devices enter or exit these zones. This technology enhances drone security by setting safe flight paths and supports context-sensitive interactions in mixed-reality metaverse scenarios through virtual perimeters. The system performance relies on precise localization to ensure timely actions within geofence boundaries.

Alternatively, vision-based security enforcement utilizes methods like visual odometry (VO), simultaneous localization and mapping (SLAM), and optical flow, often augmented with deep neural networks [22] or LiDAR [23] to enhance performance. However, vision-based techniques involve higher computational and hardware costs and suffer from reduced accuracy in visually impaired environments.

Optimal Placement: Optimizing anchor deployment for indoor tracking with ranging-based measurements is a critical aspect of wireless network localization [24]–[27]. This research focuses on achieving ubiquitous coverage within designated indoor spaces to minimize errors stemming from spatial geometry relative to users and anchors. The choice of sensor significantly influences this challenge; for example, ultrasonic sensors, due to their narrow-beam propagation, impact coverage patterns differently than RF signals, which provide omnidirectional propagation and extended range, thus requiring fewer anchors.

Addressing the challenge of achieving full coverage with a minimal number of anchors—an NP-hard problem analogous to the Art Gallery or K-Nearest problems—has been extensively explored in two-dimensional contexts [25]. However, three-dimensional coverage, essential for applications like drone geofencing systems, has received less attention. Although some studies, such as the *EGO-6* system by Famili et al. [13], have investigated the minimum number of tracking anchors required for three-dimensional geofencing coverage, they have often neglected how strategic beacon placement affects tracking accuracy.

Unlike much of the existing literature, *MetaFence* specifically focuses on the impact of anchor placement on the final 3D tracking accuracy. We consider it crucial to not only know the required number of anchors for a specific location but also to determine the most advantageous placement to minimize the 3D tracking errors caused by the spatial geometry between the user and the anchors. To that end, given a known number of required anchors for full coverage, *MetaFence* proposes a meta-RL-based approach to determine the optimal placement of 5G Points. In previous work [28], Famili et al. tackled this challenge by introducing a deep Q-learning approach, termed "*DEFENCE*", which proved effective in solving the optimal placement problem. However, *DEFENCE* necessitates solving each placement task individually, which is inefficient for practical applications involving diverse environments. In contrast, our proposed method, *MetaFence*, learns a general policy that can be efficiently adapted to new tasks without the need for complete retraining. To the best of our knowledge, *MetaFence* is the first algorithm to address the spatial geometric aspects of anchor placement for geofencing systems and proposes a meta-RL-based approach to tackle this complex issue.

This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License. For more information, see https://creativecommons.org/licenses/by-nc-nd/4.0/

This article has been accepted for publication in IEEE Open Journal of the Communications Society. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/OJCOMS.2025.3531318

Learning-based Techniques: RL is a learning paradigm in which an agent learns to make optimal decisions by interacting with its environment and taking actions through trial and error. RL has proven highly effective in deriving optimal strategies across various domains, including the navigation of autonomous vehicles and achieving superhuman performance in game playing [19]. To solve the optimization problem, each learning task is modeled as a Markov Decision Process (MDP) $\mathcal{M}$, defined by the tuple $(\mathcal{S}, \mathcal{A}, P, r)$, where $\mathcal{S}$, $\mathcal{A}$ P, r represent states, actions, the transition function and rewards, respectively. In a typical RL problem, an agent interacts with its environment by taking an action a based on the current state s. In response, the environment provides a new state s' and a reward r. The transition function $P(s' \mid s, a)$ specifies the probability of transitioning from state s to state s' given action a. The objective in RL is to discover a policy π that maximizes long-term reward.

Meta-RL [20], [21], [29] enhances traditional RL by training on a distribution of MDPs across various tasks, enabling the meta-model to quickly adapt to unseen tasks. The meta-training process starts with initializing the meta-model parameters θ and involves two optimization loops: an inner loop and an outer loop. In the inner loop, the agent performs task-specific updates to the model parameters to maximize rewards within each environment. Once the inner loop updates are completed for all tasks in each learning iteration, the outer loop refines the model further. During the outer loop, the meta-model parameters are updated by minimizing the cumulative task-specific losses, thereby improving the meta-model's adaptability across different tasks.

III. RAN Slicing for Enhanced Accuracy

In 5G deployments, network slicing [30], [31] is a fundamental feature, allowing the creation of multiple virtual networks on a single physical infrastructure. Each slice is tailored to meet the specific needs of various applications, ensuring optimal performance and resource utilization. When applied to the radio access network (RAN), the network slicing concept can be leveraged to enhance localization accuracy, catering to different use cases with varying precision requirements. By creating dedicated RAN slices optimized for localization, service providers can offer solutions that cater to the unique demands of diverse applications, ranging from consumer navigation to industrial automation.

Different use cases demand different levels of localization accuracy. For instance, augmented reality (AR) applications in retail may require localization accuracy within a few meters to enhance user experience and provide relevant information based on the user's location. Conversely, industrial applications such as automated vehicles in a smart factory necessitate centimeter-level accuracy to ensure precise navigation and operation. Similarly, public safety applications, such as emergency response systems, demand highly reliable and precise localization to quickly and accurately locate individuals in need. Leveraging the concept of RAN slicing, network operators can ensure that each use case receives the appropriate level of localization accuracy and reliability while finding a balance between QoS and positioning accuracy.

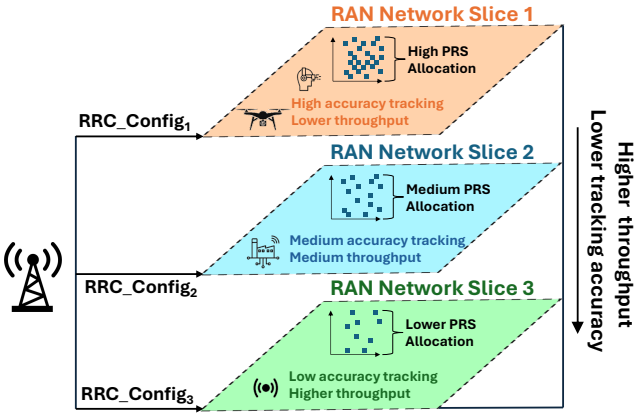


Figure 2: Tentative representation of RAN slicing to accommodate different localization requirements with varying radio resource allocation for the PRS.

Implementing RAN slicing for localization accuracy involves several technical considerations. First, the allocation of radio resources needs to be managed dynamically to ensure that each slice can meet its specific localization requirements without interfering with others. This may involve the use of advanced radio resource management (RRM) techniques to predict and allocate resources based on real-time demand and network conditions. Second, the integration of advanced localization technologies, such as enhanced GPS, triangulation, and time-of-flight measurements, is crucial to achieving the desired accuracy. Each slice can be configured with the most appropriate localization technology to optimize performance for its specific use case.

Exploiting RAN slicing allows for the customization of network performance to meet the specific needs of diverse applications, leading to improved user experiences and operational efficiencies. Moreover, it enables the efficient use of network resources by ensuring that each slice only consumes the necessary amount of resources required for its intended use case. However, there are challenges to be addressed, including the complexity of managing multiple slices with varying requirements and the need for robust security mechanisms to protect the integrity and privacy of the localization data. Additionally, ensuring seamless interoperability between different slices and legacy systems can be challenging but is essential for the widespread adoption of this technology.

IV. MetaFence Tracking Framework

In this section, we begin by outlining the essential prerequisites for localization and tracking in geofencing systems in Section A. Following this, in Section B, we describe how *MetaFence* enables real-time tracking of users through round trip time (RTT)-based trilateration. Finally, in Section C, we

This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License. For more information, see https://creativecommons.org/licenses/by-nc-nd/4.0/

This article has been accepted for publication in IEEE Open Journal of the Communications Society. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/OJCOMS.2025.3531318

provide a derivation of the CRLB, also referred to as the positioning error bound (PEB) for *MetaFence*, highlighting how the spatial geometry between the user and the 5G Points influences the overall localization error.

A. Principle Prerequisites for Tracking

For precision tracking, *MetaFence* primarily focuses on ranging-based localization techniques. These methods involve the transmission and reception of signals between a user and one or more anchor points, which are essential for accurately framing the localization problem.

The transmission mediums utilized in these localization strategies can be either acoustic [32] or RF [33], dependent on the nature of the transceivers deployed. Acoustic ultrasound transmissions, characterized by their lower propagation velocities, provide highly precise positional determinations [24]. Although they are adept at closely-spaced, high-fidelity gesture recognition scenarios [34], their utility declines across larger distances owing to their restricted reach, making them suboptimal for broad-scope geofencing initiatives.

Apart from the type of signal, the effectiveness of a localization system is dependent on measurement methodologies like received signal strength (RSS), and channel state information (CSI), among others. RSS and CSI, frequently utilized alongside environmental fingerprinting, provide an economical and simple solution. Nonetheless, tracking approaches that depend on RSS fingerprinting are significantly vulnerable to real-time changes in the environment [35], which can undermine their dependability.

AOA estimation, which utilizes angulation techniques for tracking, requires the deployment of specialized antenna arrays [36]. This technique demands substantial computational resources and necessitates the allocation of increased processing capabilities. The intricacy of this approach stems from the implementation of complex algorithms, like Multiple Signal Classification (MUSIC), which are crucial for precise angle determination.

Contrary to the complexities associated with RSS and AOA methodologies, TOA serves as a more streamlined metric for measurement [37], obviating the need for extensive computational capabilities or elaborate antenna configurations. TOA calculations are achieved by translating temporal measurements into spatial distances through the equation $d = c \times t$, where d is the spatial separation between the transmitter and receiver, c signifies the velocity of signal propagation (equivalent to the speed of light for RF signals), and t represents the time interval that RF waves require to travel from the transmitter to the receiver. By harnessing distance measurements from multiple anchor points around the user, trilateration techniques are employed for precise tracking.

B. RTT-based Spatial Mapping

In the realm of TOA-based localization and tracking, a fundamental challenge presents itself in the synchronization discrepancies between the user's clock and those of the anchor points. This misalignment results in distance measurements that are inherently biased due to synchronization errors. To address this challenge, two predominant methodologies are employed: time difference of arrival (TDOA) and time of flight (TOF). Both strategies reduce the rigorous demands for synchronization between the user and geofencing anchor points. However, the research outlined in [16] by Famili et al. indicates that TOF, particularly RTT-based measurements, provides enhanced accuracy in ranging, as corroborated by CRLB analyses. *MetaFence* capitalizes on TOA metrics derived from RTT to obtain distance measurements between 5G Points and users. For accurate tracking, trilateration requires a minimum of three anchor points in two-dimensional (2D) settings and at least four in 3D scenarios.

In the configuration of our system, the variable d_i represents the computed distance between the user, positioned at coordinates $[x\ y\ z]^T$, and each corresponding i-th 5G Point located at $[x_i\ y_i\ z_i]^T$. This relationship manifests as the equation $(x_i-x)^2+(y_i-y)^2+(z_i-z)^2 = d_i^2$ where i ranges from 1 to N, with N signifying the total number of 5G Points. To reformulate this in a more structured mathematical framework, we express the equation as $\boldsymbol{Hp} = \boldsymbol{\nu}$. Here, $\boldsymbol{H}$ is defined as $\begin{bmatrix}2(x_n - x_i) & 2(y_n - y_i) & 2(z_n - z_i)\end{bmatrix}$, and $\boldsymbol{\nu}$ is represented by $\begin{bmatrix}d_i^2 - d_n^2 - x_i^2 - y_i^2 - z_i^2 + x_n^2 + y_n^2 + z_n^2\end{bmatrix}$, where i ranges from 1 to $n-1$. In essence, the position vector of the user, symbolized by $\boldsymbol{p} = [x\ y\ z]^T$, is ascertained by resolving the equation $\boldsymbol{p} = (\boldsymbol{H}^T\boldsymbol{H})^{-1}\boldsymbol{H}^T\boldsymbol{\nu}$.

C. Positioning Error Bound

A critical benchmark in evaluating the performance of tracking technologies with a quantifiable metric is CRLB, which represents a fundamental statistical limit that defines the minimum variance achievable by an unbiased estimator.

Based on the assumption of independent range measurements that incorporate zero-mean additive Gaussian noise with a consistent variance σ_r^2, Wang et al. illustrate in [25] that within a 2D trilateration system employing an unbiased estimator, the CRLB for the variance of positional error $\sigma^2(r) = \sigma_x^2(r) + \sigma_y^2(r)$ at position r, is determined as follows:

$$\sigma(r) = \sigma_r \times \sqrt{\frac{N}{\sum_{i=1}^{N-1}\sum_{j=i+1}^{N} A_{ij}}}; \tag{1}$$

where N is the number of localization anchors, θ_i is the angle between A_i and r, A_i is the i-th anchor, and $A_{ij} = |\sin(\theta_i - \theta_j)|$.

Based on their proposed CRLB for a 2D localization system, as delineated in Equation 1, it is evident that the overall 2D localization accuracy is influenced by the accuracy of range estimation and the function $\sqrt{N(\sum_{i=1}^{N-1}\sum_{j=i+1}^{N} A_{ij})^{-1}}$. The former term ($\sigma_r$) encap-

This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License. For more information, see https://creativecommons.org/licenses/by-nc-nd/4.0/

This article has been accepted for publication in IEEE Open Journal of the Communications Society. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/OJCOMS.2025.3531318

sulates the precision in distance measurement, which may be impacted by the system's timing mechanism, the resolution of time measurements, channel noise, multipath effects, interference, and other factors. In contrast, the latter term is exclusively influenced by the relative geometry between the positioning beacons and the user.

Our aim in this work is to prioritize the examination of the latter term. While the majority of current research [36], [38]–[41] concentrates on enhancing the former (ranging errors), the significance of the latter term (errors induced by spatial geometry) and strategies for its mitigation are frequently overlooked in existing literature. To address this gap, we initially derive the CRLB for the RTT-based trilateration system utilized in the *MetaFence* tracking system. Our objective is to formulate a closed-loop expression for the PEB of the *MetaFence* tracking system, subsequently employing this formulation to address and mitigate geometry-induced errors in subsequent sections.

To begin, let's recall the tracking model configuration of the *MetaFence* system. We have a user located at $\boldsymbol{p} = [x \ y \ z]^T$, and we have N total number of 5G Points, where the i-th one is situated at $\boldsymbol{p_i} = [x_i \ y_i \ z_i]^T$. The true distance between the user and the i-th 5G Point is calculated as $d_i = \sqrt{(x-x_i)^2+(y-y_i)^2+(z-z_i)^2}$. However, the exact value of d_i is not available due to ranging measurement errors (σ_r). Therefore, we can express the ranging measurements as $r_i = d_i + n_i$, resulting in:

$$r_i = \sqrt{(x-x_i)^2+(y-y_i)^2+(z-z_i)^2} + n_i; \tag{2}$$

where n_i is the Gaussian noise affecting the i-th measurement. Without loss of generality, we can assume that the noises for different i-th ranging measurements are independent, have zero mean, and share the same variance. Therefore, we have: $n_i \sim \mathcal{N}(0, \sigma_r^2)$, where $i \in \{1, \cdots, N\}$, N is the number of 5G Points, and $\mathcal{N}(0, \sigma_r^2)$ represents Gaussian noise with zero mean and σ_r^2 variance.

The CRLB states that for any unbiased estimator $\hat{\theta}$, the variance of $\hat{\theta}$ is at least as large as the inverse of the Fisher Information Matrix (FIM):

$$\text{Var}(\hat{\theta}) \geq \mathcal{I}(\theta)^{-1}; \tag{3}$$

where θ is the parameter being estimated, and $\mathcal{I}$ is the FIM for θ. The FIM, denoted as $\mathcal{I}(\theta)$ is computed by taking the expected value of the negative second derivative (Hessian matrix) of the log-likelihood function with respect to the parameter θ. The FIM for the tracking estimator in our proposed *MetaFence* system provides a measure of the amount of information that the noisy ranging measurements carry about the unknown parameters of the user coordinates. In this case, for a vector of parameters $\boldsymbol{p} = [x \ y \ z]^T$, the FIM is given by:

$$\mathcal{I}(\boldsymbol{p}) = \mathbb{E}\left[-\frac{\partial^2 \log L(\boldsymbol{p})}{\partial \boldsymbol{p} \partial \boldsymbol{p}^T}\right]. \tag{4}$$

Alternatively, if $L(\boldsymbol{p})$ is the likelihood function, and $\mathcal{L} = \log L(\boldsymbol{p})$ is the log-likelihood, then the FIM can also be expressed as:

$$\mathcal{I}(\boldsymbol{p}) = \mathbb{E}\left[\left(\frac{\partial \mathcal{L}(\boldsymbol{p})}{\partial \boldsymbol{p}}\right)\left(\frac{\partial \mathcal{L}(\boldsymbol{p})}{\partial \boldsymbol{p}}\right)^T\right]. \tag{5}$$

The log-likelihood function for the measurements is proportional to:

$$\mathcal{L}(\boldsymbol{p}) = -\sum_{i=1}^{N} \frac{(r_i - \sqrt{(x-x_i)^2+(y-y_i)^2+(z-z_i)^2})^2}{2\sigma_r^2}.$$

For the tracking system of *MetaFence* with a vector of $\boldsymbol{p} = [x \ y \ z]^T$ and an unbiased estimator of $\hat{\boldsymbol{p}}$, the inequality in Equation 3 involving the CRLB and the FIM for an unbiased estimator of a scalar parameter $\hat{\theta}$, can be alternatively written as:

$$\text{Cov}(\hat{\boldsymbol{p}}) \geq \mathcal{I}(\boldsymbol{p})^{-1} = \begin{bmatrix} I_{xx} & I_{xy} & I_{xz} \\ I_{yx} & I_{yy} & I_{yz} \\ I_{zx} & I_{zy} & I_{zz} \end{bmatrix}^{-1}. \tag{6}$$

Taking the derivatives, we compute the entries of $\mathcal{I}(\boldsymbol{p})$ for the tracking estimator in *MetaFence* as follows:

$$\begin{aligned}
I_{xx} &= \sum_{i=1}^{N} \frac{1}{\sigma_r^2}\left(\frac{\partial \hat{r}_i}{\partial x}\right)^2, \\
I_{yy} &= \sum_{i=1}^{N} \frac{1}{\sigma_r^2}\left(\frac{\partial \hat{r}_i}{\partial y}\right)^2, \\
I_{zz} &= \sum_{i=1}^{N} \frac{1}{\sigma_r^2}\left(\frac{\partial \hat{r}_i}{\partial z}\right)^2, \\
I_{xy} = I_{yx} &= \sum_{i=1}^{N} \frac{1}{\sigma_r^2}\left(\frac{\partial \hat{r}_i}{\partial x}\right)\left(\frac{\partial \hat{r}_i}{\partial y}\right), \\
I_{xz} = I_{zx} &= \sum_{i=1}^{N} \frac{1}{\sigma_r^2}\left(\frac{\partial \hat{r}_i}{\partial x}\right)\left(\frac{\partial \hat{r}_i}{\partial z}\right), \\
I_{yz} = I_{zy} &= \sum_{i=1}^{N} \frac{1}{\sigma_r^2}\left(\frac{\partial \hat{r}_i}{\partial y}\right)\left(\frac{\partial \hat{r}_i}{\partial z}\right);
\end{aligned} \tag{7}$$

where $\hat{r_i} = d_i = \sqrt{(x-x_i)^2+(y-y_i)^2+(z-z_i)^2}$ represents the estimated distance in the absence of noise, and the partial derivatives are as follows:

$$\frac{\partial \hat{r}_i}{\partial x} = \frac{x-x_i}{\hat{r}_i}, \qquad \frac{\partial \hat{r}_i}{\partial y} = \frac{y-y_i}{\hat{r}_i}, \qquad \frac{\partial \hat{r}_i}{\partial z} = \frac{z-z_i}{\hat{r}_i}. \tag{8}$$

We can then write FIM as follows:

$$\mathcal{I}(\boldsymbol{p}) = \frac{1}{\sigma_r^2}\mathbf{A}^T\mathbf{A}; \tag{9}$$

where the matrix $\mathbf{A}$ constructed using the partial derivatives as given below:

$$\mathbf{A} = \begin{bmatrix} \frac{\partial \hat{r}_1}{\partial x} & \frac{\partial \hat{r}_1}{\partial y} & \frac{\partial \hat{r}_1}{\partial z} \\ \frac{\partial \hat{r}_2}{\partial x} & \frac{\partial \hat{r}_2}{\partial y} & \frac{\partial \hat{r}_2}{\partial z} \\ \vdots & \vdots & \vdots \\ \frac{\partial \hat{r}_N}{\partial x} & \frac{\partial \hat{r}_N}{\partial y} & \frac{\partial \hat{r}_N}{\partial z} \end{bmatrix} = \begin{bmatrix} \frac{x-x_1}{\hat{r}_1} & \frac{y-y_1}{\hat{r}_1} & \frac{z-z_1}{\hat{r}_1} \\ \frac{x-x_2}{\hat{r}_2} & \frac{y-y_2}{\hat{r}_2} & \frac{z-z_2}{\hat{r}_2} \\ \vdots & \vdots & \vdots \\ \frac{x-x_N}{\hat{r}_N} & \frac{y-y_N}{\hat{r}_N} & \frac{z-z_N}{\hat{r}_N} \end{bmatrix}. \tag{10}$$

This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License. For more information, see https://creativecommons.org/licenses/by-nc-nd/4.0/

This article has been accepted for publication in IEEE Open Journal of the Communications Society. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/OJCOMS.2025.3531318

The CRLB states that the covariance matrix $\mathbf{C}$ for our unbiased tracking estimator $\hat{\boldsymbol{p}}$ of the parameter $\boldsymbol{p} = [x\ y\ z]^T$ is lower bounded by the inverse of the FIM:

$$\mathbf{C} = \begin{bmatrix} \sigma_x^2 & \sigma_{xy} & \sigma_{xz} \\ \sigma_{yx} & \sigma_y^2 & \sigma_{yz} \\ \sigma_{zx} & \sigma_{zy} & \sigma_z^2 \end{bmatrix} = \mathcal{I}^{-1}(\boldsymbol{p}); \tag{11}$$

where the diagonal entries of $\mathbf{C}$ provide the lower bounds on the variances of the estimators of x, y, and z. Combining this with Equation 9, we have:

$$\mathbf{C} = \mathcal{I}^{-1}(\boldsymbol{p}) = (\frac{1}{\sigma_r^2}\mathbf{A}^T\mathbf{A})^{-1}. \tag{12}$$

The positioning error bound (PEB) is an application-specific form of the CRLB used in tracking and localization to determine the best achievable accuracy in estimating the position of a user. PEB is a scalar value that represents the lower bound on the Root Mean Squared Error (RMSE) of the position estimate. It is computed from the diagonal elements of the covariance matrix $\mathbf{C}$ as follows:

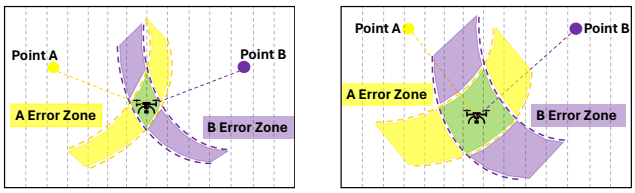
$$\text{PEB} = \sqrt{\text{trace}(\mathbf{C})}. \tag{13}$$

Incorporating Equation 11 and Equation 12 into the analysis, we derive the following results:

$$\text{PEB} = \sqrt{\sigma_x^2 + \sigma_y^2 + \sigma_z^2} = \sqrt{\text{trace}\left((\frac{1}{\sigma_r^2}\mathbf{A}^T\mathbf{A})^{-1}\right)}. \tag{14}$$

Consequently, the PEB is determined by the product of two distinct factors: the standard deviation of the ranging error, σ_r, and the summation of the diagonal elements of the square root of the inverse of the matrix $(\mathbf{A}^T\mathbf{A})$, denoted as $\sqrt{\text{trace}((\mathbf{A}^T\mathbf{A})^{-1})}$. The former, σ_r, is subject to variability introduced by several parameters that affect distance measurement accuracy. These parameters range from the design characteristics of the signal used for measurements—such as frequency and bandwidth—to the techniques employed for timing resolution, and the properties of the transmission channel, including noise, interference, and multipath effects that complicate signal reception. To quantify the impact of these factors on σ_r, we can derive the CRLB for a specific scenario. For instance, consider the application of a 5G new radio (NR) positioning reference signal (PRS) with OFDM modulation, a bandwidth of 100 MHz, 4096 subcarriers, and a subcarrier spacing of 15 kHz, propagating through a Rician fading channel with numerous non-line-of-sight (NLOS) paths. Although a detailed evaluation of each factor's contribution to σ_r is beyond the scope of this study, it is important to note that the overall positioning accuracy depends not only on σ_r. In other words, in this work, we aim to show that given a σ_r that is affected by all the aforementioned factors, the overall positioning accuracy is not merely dependent on that.

The latter factor in the PEB formulation, $\sqrt{(\mathbf{A}^T\mathbf{A})^{-1}}$, is independent of the factors influencing σ_r. As highlighted in Equation 10, the matrix $\mathbf{A}$ is solely dependent on the user's position at a specific (x, y, z) coordinate and the spatial arrangement of the 5G Points. This term represents the contribution of spatial geometry to the PEB. The function $g_x(x, y, z)$ quantifies the impact of geometric errors at the point (x, y, z) on the variance of the x-coordinate location estimation, σ_x^2, which corresponds to the first diagonal element of $\sqrt{(\mathbf{A}^T\mathbf{A})^{-1}}$. Similarly, $g_y(x, y, z)$ and $g_z(x, y, z)$ describe the influence on the y-coordinate and z-coordinate estimations, respectively, represented by the subsequent diagonal elements. In a 2D scenario, this geometric effect is denoted as $g_{2D}(x, y, z)$, whereas in a full 3D context, it is expressed as $g_{3D}(x, y, z)$.



(a) 5G Points arrangement with low tracking error

(b) 5G Points arrangement with high tracking error

Figure 3: Illustration of the impact of $g_{2D}(x, y, z)$ on tracking error resulting from varying 5G Points deployments.

During the preliminary analysis phase, we observed that the values of $g_x(x, y, z)$ and $g_y(x, y, z)$ are typically similar and generally low. This observation implies that $g_{2D}(x, y, z)$ is minimal, suggesting that spatial geometry has a limited impact on localization accuracy in 2D scenarios. It appears that the majority of the existing work focuses on 2D tracking; hence, they may have neglected the influence of spatial geometry, focusing instead on enhancing ranging-based accuracy and reducing σ_r. However, for applications such as drone geofencing and metaverse user tracking, where 3D data is crucial, this oversight becomes significant. Our analysis indicates that while the influence of the 5G Points configuration on $g_{2D}(x, y, z)$ is negligible, $g_z(x, y, z)$ is substantially affected by these configurations and typically exhibits much higher values than $g_x(x, y, z)$ and $g_y(x, y, z)$. Consequently, despite the utilization of advanced technologies for high-accuracy ranging and innovative system designs to mitigate noise and multipath effects, achieving extremely accurate distance measurements and a low σ_r may lead to excellent results in 2D tracking only. However, the accuracy of 3D tracking may still suffer significantly due to the pronounced impact of $g_{3D}(x, y, z)$ on overall 3D tracking accuracy.

Figure 3 illustrates the visual representation of $g_{2D}(x, y, z)$ for two distinct scenarios. In the first case, depicted in Figure 3a, the beacons are positioned more optimally compared to the configuration shown in Figure 3b. Consequently, this disparity in placement leads to varying location estimation errors, as indicated by the respective shaded regions.

This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License. For more information, see https://creativecommons.org/licenses/by-nc-nd/4.0/

This article has been accepted for publication in IEEE Open Journal of the Communications Society. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/OJCOMS.2025.3531318

Table 1: Evaluation of $g_{3D}(x, y, z)$ Values

$\mathbf{g_{3D}(x, y, z)}$ Values	Evaluation of 5G Points Deployment
1	Ideal
$1-2$	Very Good
$2-5$	Good
$5-10$	Medium
$10-20$	Sufficient
>20	Bad

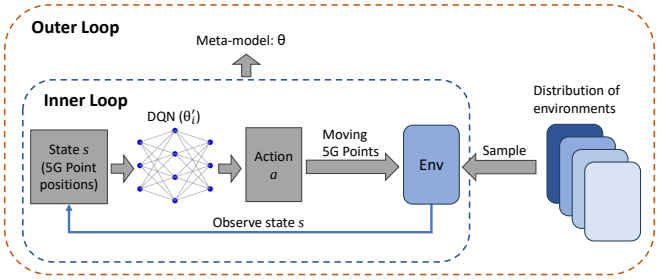


Figure 4: An overview of the meta-RL approach in *MetaFence*.

The primary goal of *MetaFence* is to devise optimal configurations for 5G Points such that the value of $g_{3D}(x, y, z)$ remains minimal across all points within the designated geofencing area, thereby minimizing the adverse effects of spatial geometry on 3D tracking accuracy and ensuring an exceptional geofencing system. To achieve this, *MetaFence* initially formulates an optimization problem aimed at identifying the most favorable arrangement of 5G Points that consistently maintains low $g_{3D}(x, y, z)$ values. Subsequently, *MetaFence* introduces a novel meta-RL strategy to address this optimization challenge, ultimately providing an effective deployment configuration for the 5G Points.

In this work, we presuppose the attainment of high-accuracy ranging capabilities, achieving a precision of 1 cm, facilitated by the expansive bandwidth provided by the PRS for 5G positioning, where $\sigma_r = 1$ cm cm. Furthermore, we stipulate that the proposed geofencing systems must maintain an overall tracking accuracy of 20 cm or better. This specification sets the upper limit for $g_{3D}(x, y, z)$ at 20. Accordingly, we have developed a chart delineating the range of $g_{3D}(x, y, z)$ values, from 1—representing the ideal scenario where the spatial configuration of the 5G Points exerts no impact on the overall 3D tracking accuracy at a specific point—to 20, a threshold that must not be exceeded. This analysis is detailed in Table 1.

V. Deployment Strategy

We formulate the placement of 5G Points as an optimization problem, in which the 5G Points are initially positioned at starting locations, as explained in Section C, and then gradually moved to achieve the lowest $\overline{g_{3D}(x, y, z)}$, where $\overline{g_{3D}(x, y, z)}$, also referred to as $\overline{g_{3D}}$, is the average of $g_{3D}(x, y, z)$ values over all the feasible user locations in the designated area. However, the varying room dimensions pose considerable challenges, as a policy learned in one environment may not be effective in another if the environments differ substantially. To address this issue, we employ meta-RL [20], [21], which involves training a general model across a diverse set of tasks, enabling quick adaptation to unseen tasks. In our scenario, the placement of 5G Points in rooms with varying dimensions is regarded as distinct tasks.

We present the fundamentals of MDP and the formulation of the optimal placement problem in Section A. The details of meta-training are presented in Section B, followed by the experimental setup and specific parameters in Section C. Figure 4 provides an overview of our meta-RL design for *MetaFence*.

A. Problem Formulation in a Markov Decision Process

For the i-th task, the corresponding MDP is denoted as $\mathcal{M}_i$, defined by the tuple $(\mathcal{S}_i, \mathcal{A}_i, P_i, r_i)$, where $\mathcal{S}_i$, $\mathcal{A}_i$, P_i, r_i represent states, actions, the transition function and rewards, respectively. The practical placement problem is formulated as follows, with the key components detailed below.

- *State Space* ($\mathcal{S}$): For the j-th 5G Point, the position is given by $\boldsymbol{p_j} = [x_j, y_j, z_j]$. Hence, the state represents the positions of the four 5G Points, denoted as $[\boldsymbol{p_1}, \boldsymbol{p_2}, \boldsymbol{p_3}, \boldsymbol{p_4}]$. As illustrated in Figure 1, we consider environments cluttered with various objects, including both static objects (e.g., tables, shelves) and dynamic objects (e.g., forklifts, drones, or people). To address the challenges posed by these obstacles and the dynamic nature of the environment, we constrain the deployment of 5G Points to positions on or near the ceiling. This approach ensures a clearer path between the anchors and the users being tracked, as the upper regions of the environment generally have less clutter and fewer obstructions.
- *Action Space* ($\mathcal{A}$): Each action moves a 5G Point one step in one of six possible directions (up, down, left, right, forward, and backward). Therefore, the action can be represented as a vector of length 4×6, corresponding to the four 5G Points. In this action vector, only one component is set to 1, while the remaining components are set to 0. For each action, the 5G Point corresponding to the active component moves in the specified direction by a predefined step size.
- *Reward Function* (r): Our objective is to minimize the $\overline{g_{3D}}$ value by strategically placing the four 5G Points. Thus, the reward function is defined as $r(s, a) = -\overline{g_{3D}}(s')$, where s is current state and s' is the subsequent state. This negative value of the $\overline{g_{3D}}$ ensures that actions leading to lower $\overline{g_{3D}}$ values, which correspond to better placement of the 5G Points, yield a higher reward.
- *Transition Function* (P): It updates the position of the specified 5G Point based on the chosen action.

This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License. For more information, see https://creativecommons.org/licenses/by-nc-nd/4.0/

This article has been accepted for publication in IEEE Open Journal of the Communications Society. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/OJCOMS.2025.3531318

Algorithm 1 Meta-Training for *MetaFence*

Outer Loop

Initialize the Q network with weights θ
Initialize replay memory $\mathcal{D}_i$ for each task $\mathcal{T}_i$
Initialize the best state s_i^* for each task $\mathcal{T}_i$
for each learning iteration **do**
  for each task $\mathcal{T}_i$ **do**
    Initialize the adapted model with $\theta_i' \leftarrow \theta$
    Run the Inner Loop to update θ_i' and obtain sampled trajectories τ_i
  end for
  Update $\theta \leftarrow \theta - \beta \nabla_\theta \sum_i \mathcal{L}_{\mathcal{T}_i}(\theta_i')$ using task-specific loss $\mathcal{L}_{\mathcal{T}_i}$ and trajectories τ_i
end for

Inner Loop

for each episode **do**
  Set s_i^* as the initial state
  for each time step t **do**
    With probability ϵ, select a random action a_t
    Otherwise select $a_t = \arg\max_a Q(s_t, a; \theta)$
    Move 5G Points according to a_t
    Store transition (s_t, a_t, r_t, s_{t+1}) in $\mathcal{D}_i$
    Sample a random mini-batch from $\mathcal{D}_i$
    Perform gradient descent to update θ_i' according to Equations 16 and 17
  end for
  Update s_i^* that achieves the lowest $\overline{g_{3D}}$
end for

Algorithm 2 Meta-Testing for *MetaFence*

Input: A new task $\mathcal{T}_{test}$ with different room dimensions; the meta-model weights θ.
Output: The optimal 5G Points' positions with the lowest $\overline{g_{3D}}$ value.

Load the model weights $\theta_{test} \leftarrow \theta$
Initialize the starting state
Update θ_{test} by running the inner loop in meta-training.
Return the best state that stores the optimal positions

B. Meta-training

We employ the Model-Agnostic Meta-Learning (MAML) algorithm within the meta-RL framework to develop a meta-model that learns to solve the optimal placement problem in various environments. For clarity, we refer to the task-specific models in the inner loop as adapted models, and the model updated in the outer loop across tasks as the meta-model. Specifically, for each task $\mathcal{T}_i$, the meta-model parameters θ are adapted to the corresponding environment within the inner loop, resulting in task-specific parameters, denoted as θ_i'. In the outer loop, multiple trajectories—sequences of states, actions, and rewards—are collected based on the adapted model parameters θ_i' from each task. The meta-model parameters θ are then updated by minimizing the cumulative task-specific losses across all tasks. The meta-training and meta-testing processes are outlined in Algorithms 1 and 2, respectively. The detailed training processes in the inner loop and outer loop are described below.

Inner loop. We utilize a deep Q-learning approach in the inner loop. This typical Q-learning algorithm evaluates the state-action pair using the Q-value $Q(s, a)$, which represents the expected value of taking an action a at a specific state s. A fundamental aspect of Q-learning is the update rule for the Q-function. This rule updates the current Q-value $Q(s, a)$ by incorporating the immediate reward $r(s, a)$ and the estimated maximum Q-value for the next state, $\max_{a'} Q(s', a')$. The update formula is expressed as follows:

$$Q_{\text{new}}(s,a) = Q(s,a) + \alpha \cdot \left[r(s,a) + \gamma \max_{a'} Q(s',a') - Q(s,a) \right]; \quad (15)$$

where α represents the learning rate for updating the Q-value, ranging from 0 to 1 to balance between prior knowledge and new information, and γ denotes the discount factor that weights immediate rewards against future rewards. Given a policy, the Q-learning method employs an ϵ-greedy algorithm with an exploration rate ϵ to select an action. With a probability of ϵ, a random action is chosen from the action space; with a probability of $1 - \epsilon$, the action that maximizes the Q-value is selected. This approach aims to balance exploration and exploitation, encouraging the agent to explore the environment more thoroughly, especially at the beginning of the training process.

Deep Q-learning employs a neural network, known as the Deep Q-Network (DQN), to approximate the Q-function. DQN takes the state as input and produces the Q-values for each possible action as output. For a task $\mathcal{T}_i$, the loss function used to update the Q-network is formulated as follows:

$$\mathcal{L}_{\mathcal{T}_i}(\theta) = \mathbb{E}_{(s,a,r,s') \sim \mathcal{T}_i} \left[\left(r + \gamma \max_{a'} Q(s', a'; \theta) - Q(s, a; \theta) \right)^2 \right]. \quad (16)$$

In the training process, the observed experiences (i.e., state, action, reward, and next state) are stored in a replay memory. The DQN is subsequently updated using randomly sampled data from this replay memory over multiple episodes.

Progressive Best-State Initialization Strategy. For the optimal placement problem, during each episode, the state that yields the highest reward (i.e., the state with the lowest $\overline{g_{3D}}$ value) is recorded as the current best state within the environment. When the environment is reset, this best state is used as the initial state for the subsequent episode. This approach is designed to tailor the RL training process to our placement problem, where the objective is not to find an optimal path from a starting point to a destination but rather to determine the destination (i.e., optimal positions).

This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License. For more information, see https://creativecommons.org/licenses/by-nc-nd/4.0/

This article has been accepted for publication in IEEE Open Journal of the Communications Society. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/OJCOMS.2025.3531318

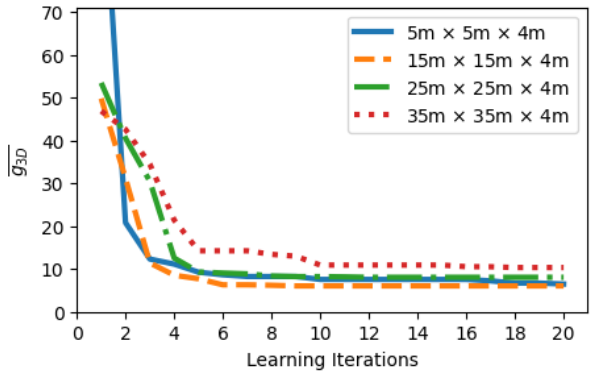


Figure 5: Convergence performance in minimizing $\overline{g_{3D}}$ across various tasks during meta-training.

By continually using the best state as the starting point, we aim to progressively improve the precision of the 5G Point placements through successive iterations of training.

In the context of meta-training, the process of adapting the meta-model parameters θ for each task $\mathcal{T}_i$ within the inner loop is represented as follows:

$$\theta_i' = \theta - \eta \nabla_\theta \mathcal{L}_{\mathcal{T}_i}(\theta); \tag{17}$$

where η is the learning rate of the inner loop. Upon the completion of each inner loop, a trajectory τ_i is collected based on the adapted model θ_i' for the outer loop update. This trajectory is generated from the initial state by taking a series of actions, observing the corresponding rewards, and transitioning to subsequent states. The trajectory is denoted as $\tau_i = \{(s_0, a_0, r_0), (s_1, a_1, r_1), ..., (s_H, a_H, r_H)\}$, where s_t, a_t and r_t represent the state, action taken, and the reward received at time step t.

Outer loop. In each learning iteration, the meta-model from the previous iteration is adapted to each task. At the end of each learning iteration, the meta-model is updated by incorporating all the adapted models and minimizing their cumulative loss based on the sampled trajectories. The outer loop update is represented as:

$$\theta \leftarrow \theta - \beta \nabla_\theta \sum_i \mathcal{L}_{\mathcal{T}_i}(\theta_i'); \tag{18}$$

where β is the learning rate of the outer loop.

Interaction Between Inner Loop and Outer Loop. The interaction between the inner and outer loops is demonstrated in Algorithm 1, specifically in Lines 6 and 9. At the start of each learning iteration, the meta-model initializes task-specific models, thereby reducing the computational complexity of inner-loop updates. Subsequently, at the end of each iteration, the outer loop refines the meta-model's parameters using the trajectories obtained from the inner loop's task-specific updates. This iterative process facilitates the generalization of the meta-model across multiple tasks.

C. Implementation of Training and Testing

In this study, we examine three approaches: the standard deep Q-learning without meta-training, the proposed meta-RL method, and an enhanced meta-RL method incorporating optimal state adaptation. The standard deep Q-learning approach, hereafter referred to as DQN, serves as the baseline in this study. Further implementation details of these three approaches are provided below.

DQN: Each task requires training the model from scratch to determine the optimal positions of 5G Points. In the absence of meta-training, the optimal placement must be solved independently for each new environment with different room dimensions. The following parameters are employed in this approach. The discount factor for updating the Q-value is set to 0.95. The Q-network is composed of three hidden layers, featuring fully connected layers with 256, 128, and 128 nodes, respectively. Each hidden layer is followed by a ReLU activation function. During training, the network weights are randomly initialized and then updated using the Adam optimizer with a learning rate of 1×10^{-4}. The maximum number of steps per episode is set to 10. The initial exploration rate is set to 1.0, with an exploration decay rate of 0.999. For the initial positioning in each environment, rather than randomly selecting their locations, the ceiling is divided into four equal grids. Each of the four 5G Points is then placed at the center of its respective grid. This approach ensures a consistent starting configuration for various tasks, thereby facilitating a more stable training process. In each step, one 5G Point is moved in the direction specified by the selected action, with a step size equivalent to 5% of the room length. Finally, the optimal state with the lowest $\overline{g_{3D}}$ value determines the positions where the 5G Points are deployed.

Meta-RL: We utilize a set of rooms with varying dimensions as the training environments, specifically with dimensions of 5 m × 5 m × 4 m, 10 m × 10 m × 4 m, 15 m × 15 m × 4 m, 20 m × 20 m × 4 m, 25 m × 25 m × 4 m, 30 m × 30 m × 4 m, and 35 m × 35 m × 4 m. In the inner loop of training for each environment, the parameters are consistent with those used in DQN. In the outer loop, the agent collects a trajectory of 10 steps from each environment to update the meta-model. The learning rate for the outer loop is set to 1×10^{-4}. During each learning iteration, each task is updated for one episode. We utilize the positioning strategy employed in DQN to initially position 5G Points for each task. During the testing phase, the trained meta-model is used as the initial model for each testing task and is further refined to achieve optimal rewards. By leveraging a diverse set of training tasks, the meta-model effectively learns and integrates information across various scenarios. As a result, during real-time deployment, the model demonstrates the ability to rapidly converge, even when handling unseen large-scale spatial configurations, thereby ensuring high efficiency.

Meta-RL with Optimal State Adaptation (meta-RL+OSA): This approach follows the training process of meta-RL, as previously described, but incorporates improve-

This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License. For more information, see https://creativecommons.org/licenses/by-nc-nd/4.0/

This article has been accepted for publication in IEEE Open Journal of the Communications Society. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/OJCOMS.2025.3531318

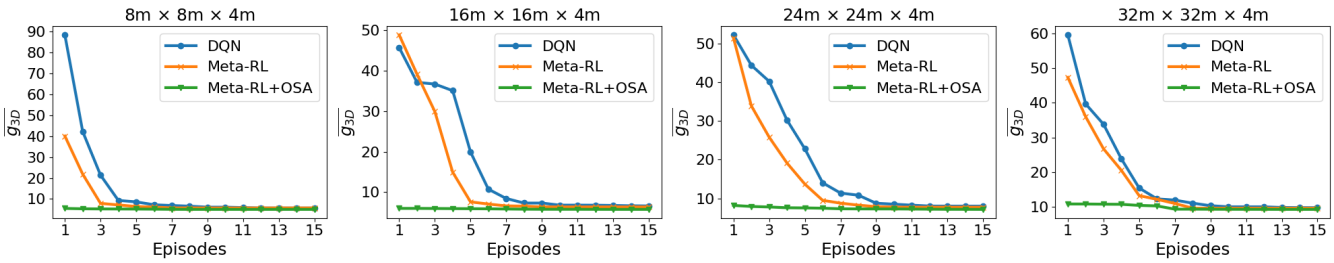


Figure 6: Comparison of the convergence performance of various RL approaches during testing.

Table 2: 5G Point configurations based on an ad-hoc placement strategy for various spaces

Space Size	Point #1	Point #2	Point #3	Point #4
8 m × 8 m × 4 m	(3,6,4)	(1.3,0.5,4)	(2.7,6.5,4)	(3,7.8,3.9)
16 m × 16 m × 4 m	(6.7,5.4,4)	(14.8,14.4,4)	(13.6,4.1,4)	(4.6,12.5,3.9)
24 m × 24 m × 4 m	(20.6,21.7,4)	(7,17.4,4)	(8.1,6.5,4)	(4,16,3.9)
32 m × 32 m × 4 m	(1.6,20,4)	(0.9,4.3,4)	(22.2,16.5,4)	(17.3,25.9,3.9)

Table 3: 5G Point configurations with the lattice-base midpoint placement strategy for various spaces

Space Size	Point #1	Point #2	Point #3	Point #4
8 m × 8 m × 4 m	(0,4,4)	(4,0,4)	(8,4,4)	(4,8,3.9)
16 m × 16 m × 4 m	(0,8,4)	(8,0,4)	(16,8,4)	(8,16,3.9)
24 m × 24 m × 4 m	(0,12,4)	(12,0,4)	(24,12,4)	(12,24,3.9)
32 m × 32 m × 4 m	(0,16,4)	(16,0,4)	(32,16,4)	(16,32,3.9)

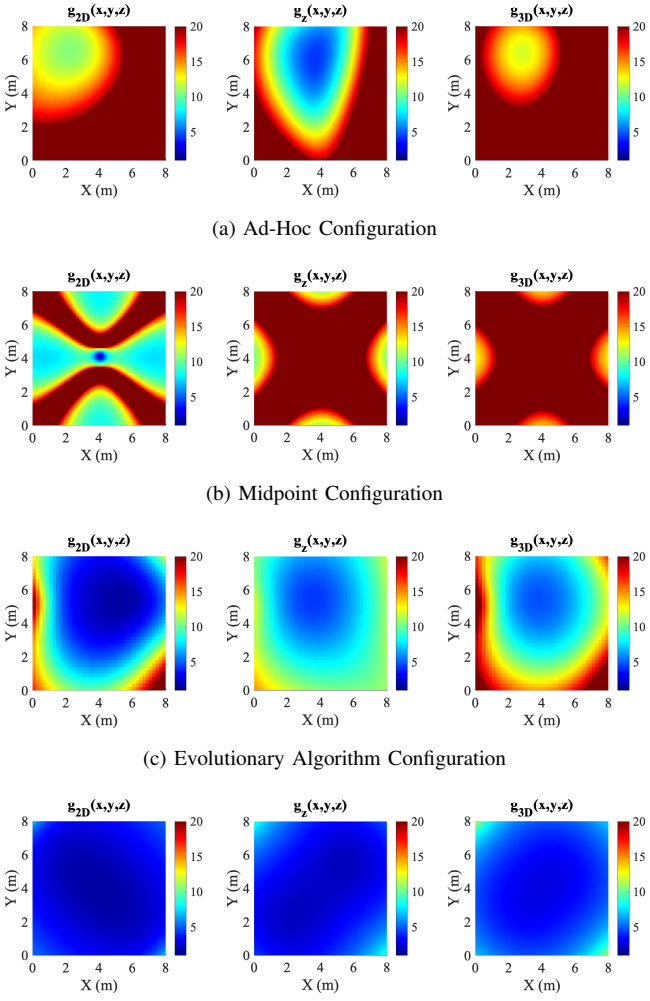
ments to enable faster adaptation and more precise positioning during the testing phase. When presented with a new task, a similar environment from the training tasks is identified. The optimal state (i.e., the positions scaled between 0 and 1) from this environment is then selected as the initial state for the testing task. Additionally, a finer-grid step size of 0.5% of the room length is utilized to further refine the model weights.

VI. Performance Evaluation

This section begins by describing our experimental framework for training the meta-RL approach and evaluating its convergence, as outlined in Sec. A. Following this, Sec. B presents an in-depth analysis of $g_{2D}(x,y,z)$, $g_z(x,y,z)$, and $g_{3D}(x,y,z)$ across different venues. Here, we compare the 5G Points configurations derived from the *MetaFence*'s meta-RL approach to those established through traditional beacon placement, underscoring the enhancements in tracking accuracy achieved by the *MetaFence* system.

A. Meta-RL Experimental Setup & Analysis

Experimental Setup. Our experiments are conducted on a virtual machine hosted on a server with an AMD EPYC 7763 processor featuring a 64-core CPU. The virtual machine, running Ubuntu 22.04.3 LTS, is allocated access to one NVIDIA A100 GPU. The detailed training parameters and



(a) Ad-Hoc Configuration

(b) Midpoint Configuration

(c) Evolutionary Algorithm Configuration

(d) *MetaFence* Configuration

Figure 7: Heatmap representation of $g_{2D}(x,y,z)$, $g_z(x,y,z)$, and $g_{3D}(x,y,z)$ values, arranged from left to right in each row, depicting various placement techniques for a small-sized space (e.g., personal private rooms) with dimensions of 8 m × 8 m × 4 m. Each panel compares traditional techniques with the *MetaFence* solution: (a) Ad-Hoc placement, as detailed in Table 2; (b) Lattice-based placement at midpoints of vertices, as illustrated in Table 3; (c) Heuristic approach using an advanced evolutionary algorithm, shown in Table 4; (d) Optimal solution proposed by *MetaFence*.

This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License. For more information, see https://creativecommons.org/licenses/by-nc-nd/4.0/

This article has been accepted for publication in IEEE Open Journal of the Communications Society. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/OJCOMS.2025.3531318

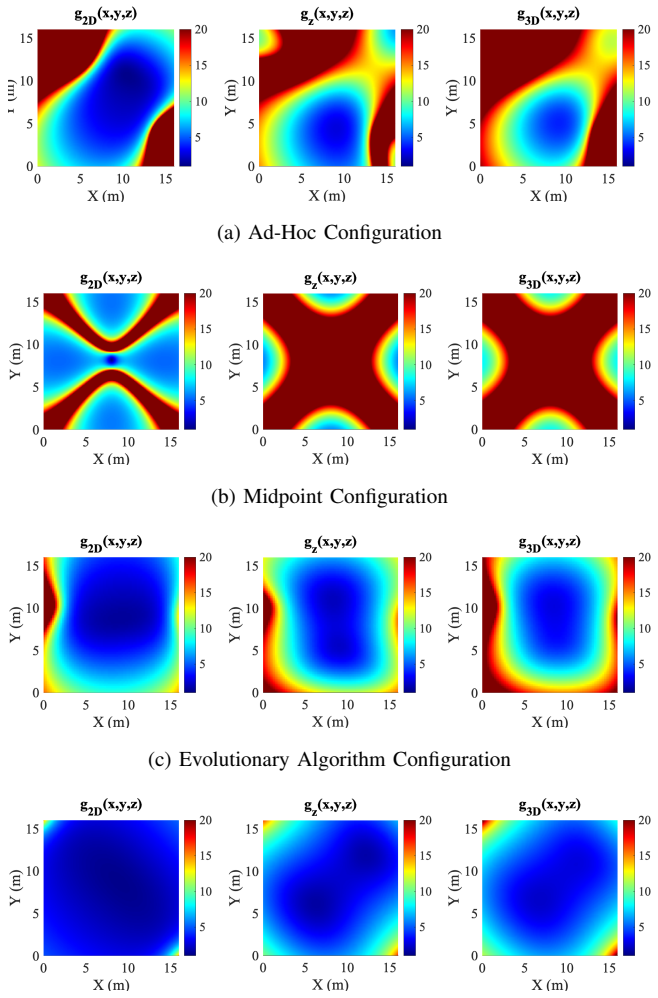
Table 4: 5G Point configurations based on the heuristic evolutionary algorithm placement strategy presented in [16] for various spaces

Space Size	Point #1	Point #2	Point #3	Point #4
8 m × 8 m × 4 m	(1.7,5.2,3.5)	(3.3,5.3,4)	(6.8,3,3.8)	(7,7.5,3.8)
16 m × 16 m × 4 m	(9,6.3,3.7)	(8.2,10.5,4)	(2.3,10.1,3.9)	(15.6,9.1,4)
24 m × 24 m × 4 m	(22,23.7,3.8)	(6.5,2.4,3.8)	(14.1,18.3,3.5)	(15.9,12.4,3.6)
32 m × 32 m × 4 m	(12,18.6,3.6)	(1.8,31.4,3.6)	(2.9,8.2,3.9)	(29.2,22.4,3.9)

testing strategies for the three approaches—DQN, Meta-RL, and Meta-RL+OSA—are outlined in Section C. We evaluate the three approaches on four new tasks that are not included in the training set. The dimensions of the rooms for these tasks are as follows: 8 m × 8 m × 4 m, 16 m × 16 m × 4 m, 24 m × 24 m × 4 m, and 32 m × 32 m × 4 m. A detailed explanation of the selection of these room dimensions will be provided in the following subsection.

Convergence in meta-training. Figure 5 illustrates the convergence performance across different environments during the meta-training process. To maintain clarity, only a subset of training tasks is displayed. Although these tasks are trained concurrently, their convergence performance is shown separately for better visualization. In this context, a learning iteration refers to an iteration within the outer loop, during which the task-specific models for each environment are updated via the inner loops. The $\overline{g_{3D}}$ value represents the value corresponding to the current best state in each learning iteration. As shown in Figure 5, tasks within different environments exhibit varying convergence rates. Instead of using rewards, this figure employs the metric $\overline{g_{3D}}$, as our objective is to minimize the $\overline{g_{3D}}$ value through the placement of 5G Points. Typically, solving the optimal placement problem in larger rooms requires more learning iterations to achieve convergence.

Fast learning in testing. As outlined in Section C, we evaluate three approaches during the testing phase: DQN, meta-RL, and meta-RL+OSA. DQN serves as the baseline, solving different tasks independently. Figure 6 illustrates the performance of these approaches on four new tasks with room dimensions that are not included in the training set. The results indicate that meta-RL outperforms DQN by achieving superior states in each epoch and reaching the optimal state more rapidly. This improvement is attributed to the meta-training process, which generates a well-generalized Q-network that serves as an effective initialization model for new tasks. Although the meta-RL approach can achieve lower $\overline{g_{3D}}$ values with fewer epochs, it still requires a gradual transition from the initial state to the optimal state. Meta-RL+OSA further accelerates the convergence speed by initializing with the best state from a similar task in the training set. This approach enables the RL agent to start from a near-optimal state and, by using a more fine-grained step size, achieve a more precise placement with a lower $\overline{g_{3D}}$ value compared to both meta-RL and DQN. For example, in



(a) Ad-Hoc Configuration

(b) Midpoint Configuration

(c) Evolutionary Algorithm Configuration

(d) *MetaFence* Configuration

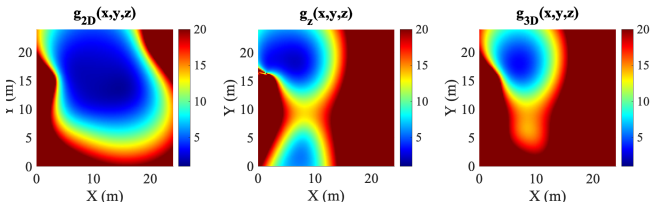
Figure 8: Heatmap representation of $g_{2D}(x, y, z)$, $g_z(x, y, z)$, and $g_{3D}(x, y, z)$ values, sequentially arranged from left to right in each row, showcasing various placement techniques for a medium-sized space (e.g., conference room) with dimensions of 16 m × 16 m × 4 m. Each panel contrasts traditional techniques with the *MetaFence* solution: (a) Ad-Hoc placement, as specified in Table 2; (b) Lattice-based placement at the midpoints of vertices, as depicted in Table 3; (c) Heuristic approach using an advanced evolutionary algorithm, as presented in Table 4; (d) Optimal solution implemented by *MetaFence*.

the environment with a dimension of 16 m × 16 m × 4 m, the DQN approach achieves a $\overline{g_{3D}}$ value of 6.55. In contrast, the meta-RL and meta-RL+OSA approaches further improve performance by reducing the $\overline{g_{3D}}$ value to 6.32 and 5.70, respectively.

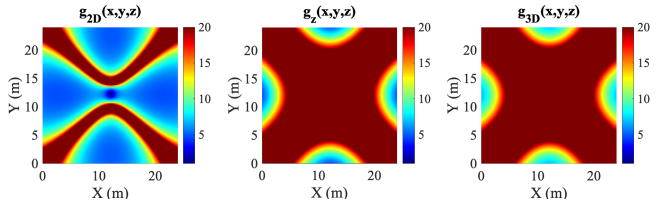
Analysis of Convergence Performance. The meta-RL+OSA method serves as the primary result of our proposed approach, representing the final and optimal solution. Its effectiveness is attributed to two key factors: (1) model

This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License. For more information, see https://creativecommons.org/licenses/by-nc-nd/4.0/

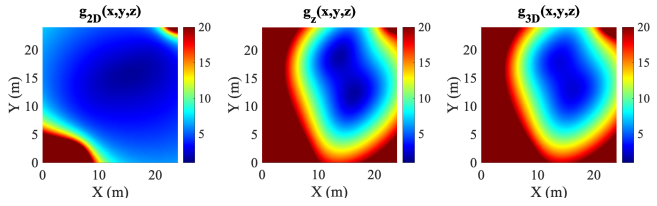
This article has been accepted for publication in IEEE Open Journal of the Communications Society. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/OJCOMS.2025.3531318



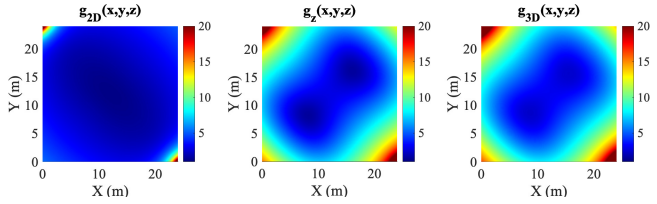
(a) Ad-Hoc Configuration



(b) Midpoint Configuration

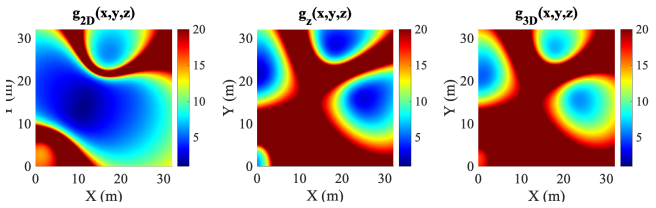


(c) Evolutionary Algorithm Configuration

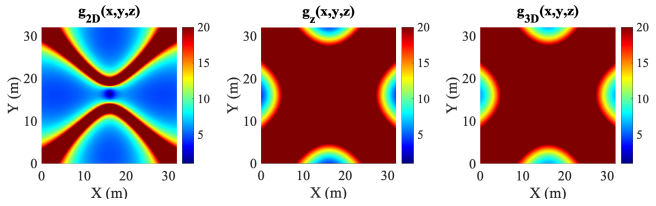


(d) *MetaFence* Configuration

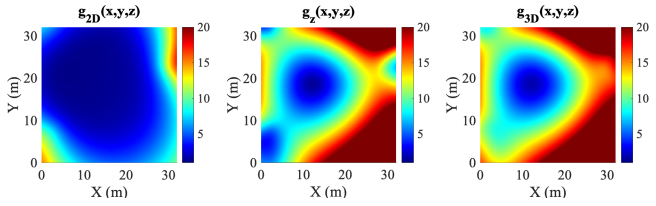
Figure 9: Heatmap representation of $g_{2D}(x, y, z)$, $g_z(x, y, z)$, and $g_{3D}(x, y, z)$ values, arranged from left to right in each row, illustrating various placement strategies for a large space (e.g., virtual reality game room) with dimensions of 24 m $\times$ 24 m $\times$ 4 m. Each panel compares traditional placement techniques with the *MetaFence* solution: (a) Ad-Hoc placement, as detailed in Table 2; (b) Lattice-based placement at the midpoints of vertices, as shown in Table 3; (c) Heuristic approach using an advanced evolutionary algorithm, as outlined in Table 4; (d) Optimal solution crafted by *MetaFence*.



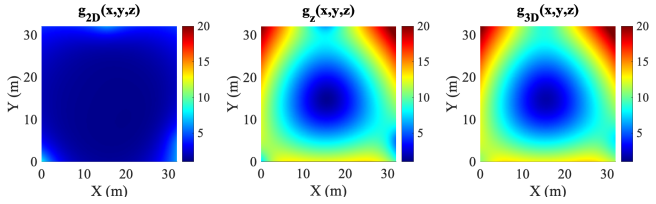
(a) Ad-Hoc Configuration



(b) Mipoint Configuration



(c) Evolutionary Algorithm Configuration



(d) *MetaFence* Configuration

Figure 10: Heatmap representation of $g_{2D}(x, y, z)$, $g_z(x, y, z)$, and $g_{3D}(x, y, z)$ values, sequentially arranged from left to right in each row, highlighting various placement strategies for an extremely large space (e.g., warehouse) with dimensions of 32 m $\times$ 32 m $\times$ 4 m. Each panel contrasts traditional placement methods with the *MetaFence* solution: (a) Ad-Hoc placement, as detailed in Table 2; (b) Lattice-based placement at the midpoints of vertices, as illustrated in Table 3; (c) Heuristic approach using an advanced evolutionary algorithm, as specified in Table 4; (d) Optimal solution implemented by *MetaFence*.

weight inheritance and (2) initial state inheritance. The model weight inheritance allows the meta-RL approach to achieve lower $\overline{g_{3D}}$ values compared to the DQN approach. The relatively small error difference is due to the use of a constant number of steps in each episode, which limits the agent's movement. Despite this constraint, the meta-RL approach demonstrates superior adaptability by enabling the Q-network to obtain better model weights. The initial state inheritance addresses this limitation by using the meta-RL+OSA method, which initializes the Q-network with model weights trained in a prior environment. This approach positions the agent closer to the optimal state, allowing it to reach the optimal solution more efficiently, thereby enhancing overall performance. In addition, previous research [42] has employed deep reinforcement learning (DRL) approaches, such as Proximal Policy Optimization (PPO), which combine various training tasks into a single optimization problem. While effective, these methods typically

This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License. For more information, see https://creativecommons.org/licenses/by-nc-nd/4.0/

This article has been accepted for publication in IEEE Open Journal of the Communications Society. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/OJCOMS.2025.3531318

require substantial computational resources and thousands of training steps. Furthermore, their real-time adaptability to dynamic changes in indoor environmental conditions is often limited. In contrast, our proposed algorithm is designed for greater efficiency. By initializing each episode with the optimal state from the preceding episode, the agent rapidly converges to a suboptimal region and then gradually refines its position to an optimal state, significantly reducing computational overhead while maintaining robust performance.

B. MetaFence Overall Evaluation

To assess the overall effectiveness of *MetaFence*, we configured a comprehensive test setup utilizing MATLAB 2024a on an Apple MacBook Pro with an M3 Max CPU and 64 GB of RAM. The primary objective is to show that, upon completing its training phase, *MetaFence* is capable of identifying the optimal configuration of 5G Points for any given area of interest that requires the implementation of a geofencing system. As this task depends heavily on the specific dimensions of the venue, especially the provided floor plan, we demonstrate the algorithm's capability across different room sizes to ensure its adaptability to various environments.

We tested four setups with dimensions ranging from $8\text{ m} \times 8\text{ m} \times 4\text{ m}$ to $32\text{ m} \times 32\text{ m} \times 4\text{ m}$, covering a spectrum from small to extremely large venues, to evaluate the algorithm's performance across different scales. While smaller spaces typically facilitate good performance, even in traditional heuristic algorithms due to simpler problem constraints, our results indicate that *MetaFence* not only performs well in these conditions but also thrives in much larger environments. Here, traditional algorithms—such as evolutionary algorithms, genetic algorithms, and simulated annealing algorithms—often struggle as the complexity and size of the NP-hard problem significantly increase computational demands, demonstrating *MetaFence*'s superior adaptability and efficiency. Moreover, given the geofencing system's use case, designed to safeguard drones and metaverse users, it is essential to cater to a variety of venue sizes. This includes compact spaces typical of personal rooms, measuring $8\text{ m} \times 8\text{ m} \times 4\text{ m}$, to medium-sized settings such as offices and conference venues at $16\text{ m} \times 16\text{ m} \times 4\text{ m}$, larger areas like expansive virtual game rooms with dimensions of $24\text{ m} \times 24\text{ m} \times 4\text{ m}$, up to extremely large spaces such as warehouses, which measure $32\text{ m} \times 32\text{ m} \times 4\text{ m}$. It is important to note that since the dimensions of $5\text{ m} \times 5\text{ m} \times 4\text{ m}$, $15\text{ m} \times 15\text{ m} \times 4\text{ m}$, $25\text{ m} \times 25\text{ m} \times 4\text{ m}$, and $35\text{ m} \times 35\text{ m} \times 4\text{ m}$ were used during the training phase of our algorithm, we excluded these sizes from our testing phase. Instead, we conducted tests on the new venues to ensure a robust evaluation of the system's performance across a range of scenarios. Furthermore, in scenarios where venue dimensions exceed $35\text{ m} \times 35\text{ m} \times 4\text{ m}$ where the 5G Points' coverage does not fully span the floor plan, the solution is simply to install additional 5G Points. This adjustment guarantees adequate coverage, allowing *MetaFence* to effectively identify the optimal placements for these new 5G Points.

Table 5: 5G Point configurations with *MetaFence* solution for various spaces

Space Size	Point #1	Point #2	Point #3	Point #4
8 m × 8 m × 4 m	(2.6,2.6,4)	(5.1,5.1,4)	(0,8,3.6)	(8,0,3.5)
16 m × 16 m × 4 m	(6.6,6.4,4)	(11.5,11.2,3.9)	(0,16,3.6)	(16,0,3.5)
24 m × 24 m × 4 m	(8.6,8.63.9)	(15.5,15.6,4)	(0,24,3.6)	(24,0,3.6)
32 m × 32 m × 4 m	(0,0,3.5)	(15.4,3,3.6)	(15.5,14.6,4)	(32,5.3,3.5)

To assess *MetaFence*'s performance, we utilize heatmaps to illustrate the $g(x, y, z)$ values across all the points in any given venue. These heatmaps detail the $g_{2D}(x, y, z)$, $g_z(x, y, z)$, and $g_{3D}(x, y, z)$ metrics within 3D spaces. Given the necessity of a fourth dimension to represent these values, traditional heatmaps struggle with clarity as data points overlap. To overcome this, we average the $g(x, y, z)$ values for each (x, y) coordinate across all z levels and display these averages in our heatmaps. This method enhances the spatial representation and enables visualization on all z planes, offering a more thorough analysis.

To benchmark our results, we conducted comparisons with three state-of-the-art placement techniques described in the literature [25], [27], [38], [41], [43]–[48]. Initially, we evaluated against traditional ad-hoc placement. This placement is particularly advantageous because it does not require high computational power and can be deployed in any dimension. However, a significant downside is that it does not account for errors induced by spatial geometry, as shown in the following figures. As a result, while this technique may be suitable for 2D localization and tracking problems where geometry has less impact, it performs poorly in 3D scenarios, significantly affecting the system's final performance. Secondly, we considered traditional lattice-based placement [38], [41], where sensors are strategically positioned at midpoints along the sides between vertices. Lastly, we compared our results with advanced heuristic algorithms, specifically utilizing the enhanced evolutionary algorithm as used in [16], [24], [32] and adapting it to our scenarios. Here, the mechanism of this evolutionary algorithm is briefly described as follows: it begins with random placements, and through various iterations, it selects better placements based on the fitness function, eliminates the less effective ones, and generates new placements using the current favorable placements as parents. The advantage of this approach as we will demonstrate in future figures, is its ability to achieve good results relatively quickly in smaller dimensions, making it superior to ad-hoc or midpoint lattice-based techniques. However, as the problem dimension increases, these algorithms significantly slow down, even with high computational resources, and as we will illustrate in subsequent figures, even with long time and high computational power, their overall performance diminishes compared to the smaller dimensions. To ensure a fair comparison,

This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License. For more information, see https://creativecommons.org/licenses/by-nc-nd/4.0/

This article has been accepted for publication in IEEE Open Journal of the Communications Society. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/OJCOMS.2025.3531318

we replicated the placements under the same conditions and environments used for our *MetaFence* system. The placements for each technique in various spaces are detailed in Table 2, Table 3, and Table 4, respectively.

We illustrate the $g_{2D}(x, y, z)$, $g_z(x, y, z)$, and $g_{3D}(x, y, z)$ values across four different spaces where: Figure 7 $\rightarrow 8 \times 8$, Figure 8 $\rightarrow$ 16 $\times$ 16, Figure 9 $\rightarrow$ 24 $\times$ 24, and Figure 10 $\rightarrow$ 32 $\times$ 32 designate the respective room dimensions used to compare our proposed *MetaFence* system with existing strategies. Throughout Figures 7 - 10, the rows (a) - (d) correspond to the ad-hoc placement, midpoint placement, placements using the enhanced evolutionary algorithm, and finally *MetaFence*, respectively

Each row's first column displays the g_{2D} values, indicating horizontal accuracy on the $X-Y$ plane. The middle column shows the g_z values for vertical accuracy along the Z-axis, and the third column presents the g_{3D} values, reflecting overall 3D accuracy for each floor plan. These figures demonstrate that while traditional placement strategies often provide satisfactory results in 2D scenarios, they tend to perform poorly in estimating vertical dimensions. This is evident in the figures, which show that vertical measurements substantially affect the overall 3D accuracy. Originally, the focus in localization literature has been on 2D accuracy, which may explain the lack of attention to the detrimental impacts of spatial geometry on 3D accuracy. However, with the growing importance of vertical accuracy for applications like drones and metaverse environments, the negative effects of spatial geometry on overall accuracy have become more apparent and critical to address. This shift underscores the need for our investigation into these effects and our proposal of solutions to mitigate their impact.

The row (d) throughout Figures 7 - 10 shows the performance of the *MetaFence* system as detailed in Table 5. This visualization reveals that *MetaFence* consistently provides optimal configurations that enhance both 2D and 3D spatial estimations across all room sizes. The heatmaps utilize red to signify poor geometric configurations that negatively impact tracking, and blue to indicate low $g(x, y, z)$ values, suggesting minimal adverse effects on tracking accuracy. Notably, *MetaFence*'s solutions primarily display non-red hues, showcasing their superior performance in various dimensions, in contrast to traditional methods that often struggle, especially with vertical and overall 3D accuracy.

To provide a quantitative analysis of the $g(x, y, z)$ values and facilitate a precise comparison with traditional benchmarks, we have generated cumulative distribution function (CDF) plots, presented in Figure 11. The first row (Figure 11a - Figure 11d) illustrates the ad-hoc benchmark, based on placements from Table 2. The second row (Figure 11e - Figure 11h) displays the lattice-based midpoint benchmark from Table 3. The third row (Figure 11i - Figure 11l) features results from a heuristic algorithm using the evolutionary approach proposed in [16], with placements provided in Table 4. The final row (Figure 11m - Figure 11p) showcases results from our *MetaFence* system, with configurations detailed in Table 5. Each row progresses from results for small (8 m $\times$ 8 m $\times$ 4 m) to expansively large (32 m $\times$ 32 m $\times$ 4 m) room dimensions, left to right.

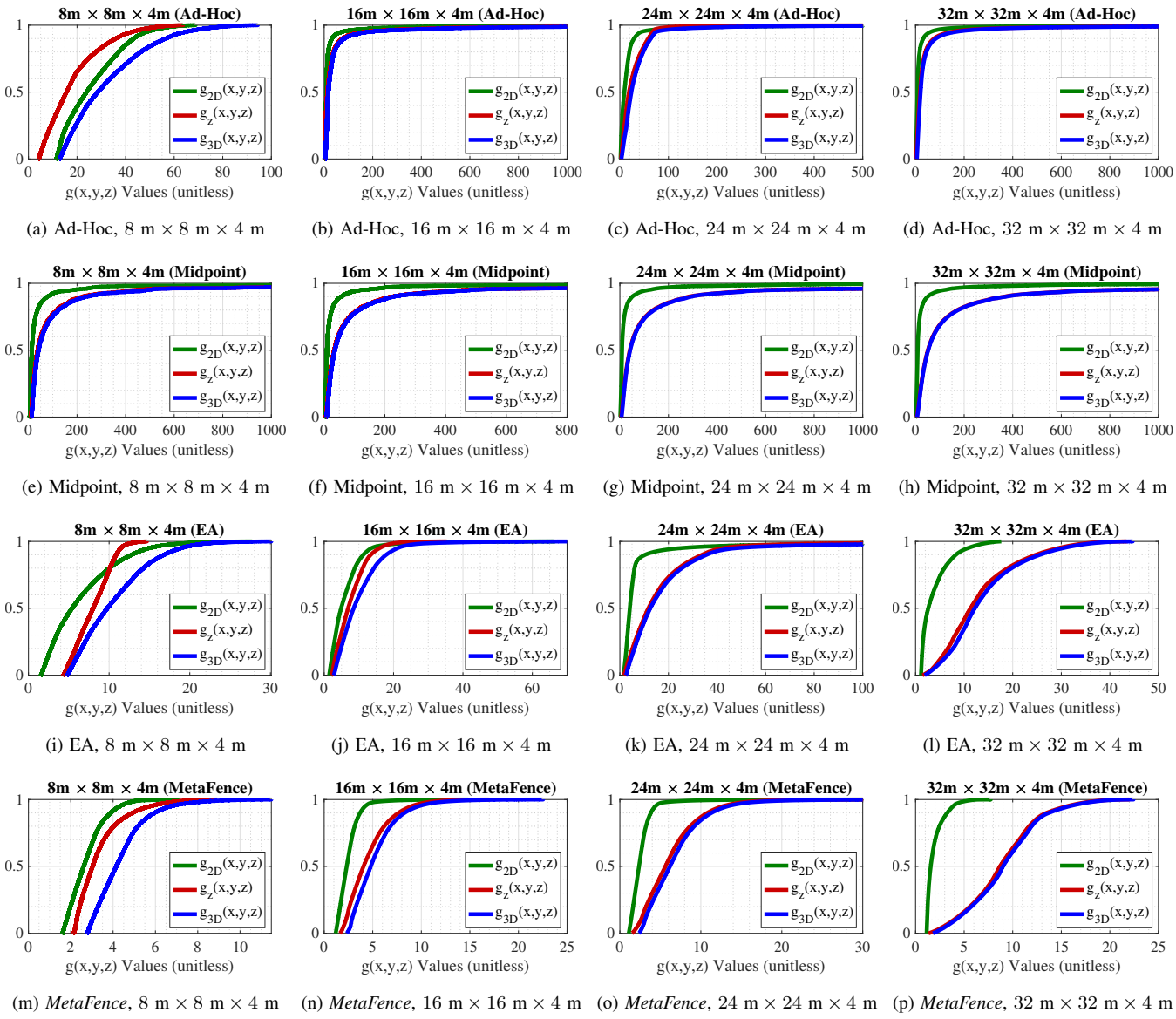
The primary objective of *MetaFence* was to assess the impact of spatial geometry of 5G Points on 3D tracking accuracy and to develop a method for optimizing their placement to mitigate this effect. As demonstrated in Figure 11m - Figure 11p, *MetaFence* consistently configures 5G Points to ensure that the majority of points in spaces requiring geofencing maintain $g_{3D}(x, y, z)$ values below 20, aligning with the system's design goals. Notably, as shown in Figure 11m, *MetaFence* achieves even better results in small rooms (8 m $\times$ 8 m $\times$ 4 m dimensions), keeping the majority of $g_{3D}(x, y, z)$ values below 10. In larger rooms (16 m $\times$ 16 m $\times$ 4 m dimensions), as depicted in Figure 11n, most $g_{3D}(x, y, z)$ values are kept under 15, and this trend continues with values under 20 in even larger spaces, according to Figure 11o and Figure 11p. Overall, *MetaFence* effectively meets its foundational goal of maintaining the majority of tracking values under the threshold of 20, illustrating its robust capability across various dimensions.

The comparative analysis of state-of-the-art geofencing techniques as depicted in Figures 11a through 11h reveals that ad-hoc and traditional lattice-based strategies, specifically those maintaining 5G Points at midpoints between vertices as discussed in [38], [41], fall short in effectiveness. These methods see $g_{3D}(x, y, z)$ values escalating to the thousands, more than 50 times higher than those achieved by the *MetaFence* solution, severely compromising tracking accuracy. Consequently, such methods prove unsuitable for reliable geofencing systems. Conversely, the evolutionary algorithm presented in [16] marks a significant improvement over these conventional techniques, as evidenced from Figures 11i to 11l. This algorithm manages to maintain $g_{3D}(x, y, z)$ values below 60 across all room dimensions, even in larger settings, an important advancement given the complexity increase with spatial dimensions, which typically exacerbates the challenge of solving NP-hard problems. This contrast is stark against the poor outcomes from traditional methods shown in Figures 11a and 11h. Yet, when comparing the evolutionary algorithm with *MetaFence*, the latter consistently outperforms, maintaining $g_{3D}(x, y, z)$ values that are, on average, more than 3 times lower regardless of spatial dimensions. Such a substantial reduction underscores the superior capability of *MetaFence* to minimize the adverse effects of spatial geometry on overall 3D tracking accuracy, solidifying its standing as a highly effective geofencing solution.

To elucidate, consider a geofencing system utilizing high-precision ranging technology, with a $\sigma_r = 1$ cm, achieved by allocating significant resources and bandwidth specifically for tracking purposes. Figures 11a through 11d demonstrate that traditional ad-hoc deployment strategies may only yield a 3D tracking accuracy of approximately 10 m, calcu-

This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License. For more information, see https://creativecommons.org/licenses/by-nc-nd/4.0/

This article has been accepted for publication in IEEE Open Journal of the Communications Society. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/OJCOMS.2025.3531318



(a) Ad-Hoc, 8 m × 8 m × 4 m (b) Ad-Hoc, 16 m × 16 m × 4 m (c) Ad-Hoc, 24 m × 24 m × 4 m (d) Ad-Hoc, 32 m × 32 m × 4 m

(e) Midpoint, 8 m × 8 m × 4 m (f) Midpoint, 16 m × 16 m × 4 m (g) Midpoint, 24 m × 24 m × 4 m (h) Midpoint, 32 m × 32 m × 4 m

(i) EA, 8 m × 8 m × 4 m (j) EA, 16 m × 16 m × 4 m (k) EA, 24 m × 24 m × 4 m (l) EA, 32 m × 32 m × 4 m

(m) *MetaFence*, 8 m × 8 m × 4 m (n) *MetaFence*, 16 m × 16 m × 4 m (o) *MetaFence*, 24 m × 24 m × 4 m (p) *MetaFence*, 32 m × 32 m × 4 m

Figure 11: CDF plots of $g_{2D}(x, y, z)$, $g_z(x, y, z)$, and $g_{3D}(x, y, z)$ values across different spatial configurations and room dimensions. The first row (referenced in Table 2) illustrates results from the ad-hoc placement. The second row (referenced in Table 3) depicts the lattice-based midpoint approach. The third row showcases the heuristic approach, specifically based on the evolutionary algorithm (EA) solution proposed in [16] and detailed in Table 4. The final row presents results from the *MetaFence* optimal solution as listed in Table 5. Each row progresses through room dimensions from 8 m × 8 m × 4 m to 32 m × 32 m × 4 m, displayed from left to right across the columns.

lated from the equation $\sigma_T(x, y, z) = \sqrt{\sigma_x^2 + \sigma_y^2 + \sigma_z^2} = g_{3D}(x, y, z) \cdot \sigma_r$, thereby rendering substantial precision investments ineffective and surpassing acceptable accuracy levels. This pattern holds true for traditional lattice-based placements at midpoints between vertices, which similarly exhibit large $g_{3D}(x, y, z)$ values (1 cm × 1000 = 10 m), as illustrated from Figures 11e to 11h. Conversely, Figures 11i to 11l show that employing an advanced evolutionary algorithm, as proposed in [16], can enhance overall 3D tracking accuracy to under a meter (1 cm × 60 = 60 cm), marking a significant improvement. Yet, the *MetaFence* framework for 5G Point deployment, as shown in Figures 11m to 11p, achieves even more precise tracking, with accuracies below 20 cm. This represents an improvement nearly fifty times greater than traditional strategies and three times better than the advanced heuristic placements. This significant difference underscores the critical impact of spatial geometry on tracking precision and the vital importance of

This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License. For more information, see https://creativecommons.org/licenses/by-nc-nd/4.0/

This article has been accepted for publication in IEEE Open Journal of the Communications Society. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/OJCOMS.2025.3531318

employing an algorithm like *MetaFence* to minimize these errors and ensure a robust geofencing system.

VII. Conclusion and Future Work

Conclusion: In this work, we introduce *MetaFence*, an advanced geofencing system that achieves improved tracking precision by utilizing indoor 5G small cells. By optimizing the placement of these 5G Points through an enhanced meta-RL approach, *MetaFence* addresses the NP-hard challenge of deployment optimization and significantly boosts geofencing accuracy. Our extensive testing demonstrates that *MetaFence* surpasses traditional methods and heuristic approaches in tracking accuracy, providing robust protection against unauthorized access and security breaches in both physical and digital spaces.

Future Work: In conjunction with this research, we have developed a small-scale 5G testbed for over-the-air performance evaluations [31]. We plan to integrate *MetaFence* into this testbed to create a proof-of-concept geofencing system in the future. Furthermore, enhancing the interpretability of RL models is particularly important for complex tasks such as localization [49]. As part of future work, we aim to investigate the integration of explainable RL techniques into our proposed approach. As a final note, *MetaFence* was initially designed to deploy meta-reinforcement learning for the optimal configuration of 5G Points in indoor setups, specifically to mitigate spatial geometry-induced errors affecting localization and tracking within geofencing systems. However, the core technique is versatile and can be extended to other applications. For example, it could enhance localization accuracy in smart buildings within smart cities by strategically placing 5G Points, or other types of localization anchors, based on positions calculated by our algorithm. Additionally, this approach can support autonomous driving applications, particularly in underground parking areas where GPS signals are unavailable, by optimizing the placement of 5G Points or alternative anchors like Wi-Fi routers. Although *MetaFence* was specifically developed for geofencing, its underlying principles offer broad applicability for improving localization and positioning across diverse scenarios.

References

[1] V. Sihag, G. Choudhary, P. Choudhary, and N. Dragoni, "Cyber4Drone: a systematic review of cyber security and forensics in next-generation drones," *Drones*, vol. 7, no. 7, 2023. [Online]. Available: https://www.mdpi.com/2504-446X/7/7/430

[2] N. Schiller, M. Chlosta, M. Schloegel, N. Bars, T. Eisenhofer, T. Scharnowski, F. Domke, L. Schönherr, and T. Holz, "Drone security and the mysterious case of DJI's DroneID." in *NDSS*, 2023.

[3] K. Liu, T. Zheng, T. Zhou, C. Liu, F. Liu, and Z. Cai, "The security and privacy concerns on Metaverse," in *International Artificial Intelligence Conference*. Springer, 2023, pp. 287–305.

[4] H. J. Hadi, Y. Cao, K. U. Nisa, A. M. Jamil, and Q. Ni, "A comprehensive survey on security, privacy issues and emerging defence technologies for UAVs," *Journal of Network and Computer Applications*, vol. 213, p. 103607, 2023. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1084804523000267

[5] J. Robie, A. Famili, and A. Stavrou, "Revisiting the spaceborne illuminators of opportunity for airborne object tracking," *Computer*, vol. 56, no. 1, pp. 82–92, 2023.

[6] M. Ali, F. Naeem, G. Kaddoum, and E. Hossain, "Metaverse communications, networking, security, and applications: Research issues, state-of-the-art, and future directions," *IEEE Communications Surveys & Tutorials*, vol. 26, no. 2, pp. 1238–1278, 2024.

[7] C. Kumar and S. Mohanty, "Current trends in cyber security for drones," in *2021 International Carnahan Conference on Security Technology (ICCST)*, 2021, pp. 1–5.

[8] C. Boselli, J. Danis, S. McQueen, A. Breger, T. Jiang, D. Looze, and D. Ni, "Geo-fencing to secure airport perimeter against suas," *International Journal of Intelligent Unmanned Systems*, pp. 102–116, 2017.

[9] A. Famili, A. Stavrou, H. Wang, J.-M. J. Park, and R. Gerdes, "Securing your airspace: Detection of drones trespassing protected areas," *Sensors*, vol. 24, no. 7, 2024. [Online]. Available: https://www.mdpi.com/1424-8220/24/7/2028

[10] Y. Wang, Z. Su, N. Zhang, R. Xing, D. Liu, T. H. Luan, and X. Shen, "A survey on metaverse: Fundamentals, security, and privacy," *IEEE Communications Surveys & Tutorials*, pp. 1–1, 2022.

[11] R. Di Pietro and S. Cresci, "Metaverse: Security and privacy issues," in *2021 Third IEEE International Conference on Trust, Privacy and Security in Intelligent Systems and Applications (TPS-ISA)*, 2021, pp. 281–288.

[12] A. Gupta, H. U. Khan, S. Nazir, M. Shafiq, and M. Shabaz, "Metaverse security: Issues, challenges and a viable ZTA model," *Electronics*, vol. 12, no. 2, 2023. [Online]. Available: https://www.mdpi.com/2079-9292/12/2/391

[13] A. Famili, A. Stavrou, H. Wang, and J.-M. J. Park, "EGO-6: enhancing geofencing security systems with optimal deployment of 6G TRPs," in *2023 Silicon Valley Cybersecurity Conference (SVCC)*, 2023, pp. 1–8.

[14] Y. Shevchenko and U.-D. Reips, "Geofencing in location-based behavioral research: Methodology, challenges, and implementation," *Behavior Research Methods*, pp. 1–29, 2023.

[15] J. Tan, E. Sumpena, W. Zhuo, Z. Zhao, M. Liu, and S.-H. G. Chan, "Iot geofencing for covid-19 home quarantine enforcement," *IEEE Internet of Things Magazine*, vol. 3, no. 3, pp. 24–29, 2020.

[16] A. Famili, T. O. Atalay, A. Stavrou, H. Wang, and J.-M. Park, "OFDRA: Optimal femtocell deployment for accurate indoor positioning of RIS-mounted AVs," *IEEE Journal on Selected Areas in Communications*, vol. 41, no. 12, pp. 3783–3798, 2023.

[17] N. Bulusu, J. Heidemann, and D. Estrin, "Adaptive beacon placement," in *Proceedings 21st International Conference on Distributed Computing Systems*, 2001, pp. 489–498.

[18] W. He, P.-H. Ho, and J. Tapolcai, "Beacon deployment for unambiguous positioning," *IEEE Internet of Things Journal*, vol. 4, no. 5, pp. 1370–1379, 2017.

[19] Y. Li, "Deep reinforcement learning: An overview," *arXiv preprint arXiv:1701.07274*, 2017.

[20] C. Finn, P. Abbeel, and S. Levine, "Model-agnostic meta-learning for fast adaptation of deep networks," in *International conference on machine learning (ICML)*. PMLR, 2017, pp. 1126–1135.

[21] R. Fakoor, P. Chaudhari, S. Soatto, and A. J. Smola, "Meta-q-learning," *arXiv*, 2019.

[22] R. P. Padhy, S. Verma, S. Ahmad, S. K. Choudhury, and P. K. Sa, "Deep neural network for autonomous uav navigation in indoor corridor environments," *Procedia Computer Science*, vol. 133, pp. 643 – 650, 2018, international Conference on Robotics and Smart Manufacturing (RoSMa2018). [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1877050918310524

[23] N. Jeong, H. Hwang, and E. T. Matson, "Evaluation of low-cost lidar sensor for application in indoor uav navigation," in *2018 IEEE Sensors Applications Symposium (SAS)*, March 2018, pp. 1–5.

[24] A. Famili, A. Stavrou, H. Wang, Jung-Min, and Park, "iDROP: Robust localization for indoor navigation of drones with optimized beacon placement," 2022.

[25] H. Wang, N. Rajagopal, A. Rowe, B. Sinopoli, and J. Gao, "Efficient beacon placement algorithms for time-of-flight indoor localization," in *Proceedings of the 27th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*, ser. SIGSPATIAL '19. New York, NY, USA: Association for Computing Machinery, 2019, p. 119–128. [Online]. Available: https://doi.org/10.1145/3347146.3359344

[26] A. Famili and A. Stavrou, "Eternal flying: Optimal placement of wireless chargers for nonstop drone flights," in *2022 International*

This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License. For more information, see https://creativecommons.org/licenses/by-nc-nd/4.0/

This article has been accepted for publication in IEEE Open Journal of the Communications Society. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/OJCOMS.2025.3531318

Conference on Electrical, Computer and Energy Technologies (ICECET), 2022, pp. 1–6.

[27] R. Sharma and V. Badarla, "Analysis of a novel beacon placement strategy 3d localization in indoor spaces," in *2019 11th International Conference on Communication Systems Networks (COMSNETS)*, pp. 320–327.

[28] A. Famili, S. Sun, T. O. Atalay, and A. Stavrou, "Precision tracking in geofencing systems using deep reinforcement learning," in *2024 International Performance, Computing, and Communications Conference (IPCCC)*, Orlando, USA, Nov. 2024, p. 10.

[29] J. Beck, R. Vuorio, E. Z. Liu, Z. Xiong, L. Zintgraf, C. Finn, and S. Whiteson, "A survey of meta-reinforcement learning," *arXiv*, 2023.

[30] S. Zhang, "An Overview of Network Slicing for 5G," *IEEE Wireless Communications*, vol. 26, no. 3, pp. 111–117, 2019.

[31] T. O. Atalay, A. Famili, D. Stojadinovic, and A. Stavrou, "Demystifying 5g traffic patterns with an indoor ran measurement campaign," in *GLOBECOM 2023-2023 IEEE Global Communications Conference*. IEEE, 2023, pp. 1185–1190.

[32] A. Famili, A. Stavrou, H. Wang, and J.-M. J. Park, "PILOT: high-precision indoor localization for autonomous drones," *IEEE Transactions on Vehicular Technology*, pp. 1–15, 2022.

[33] R. C. Luo and T.-J. Hsiao, "Indoor localization system based on hybrid Wi-Fi/BLE and hierarchical topological fingerprinting approach," *IEEE Transactions on Vehicular Technology*, pp. 10 791–10 806, 2019.

[34] W. Wang, A. X. Liu, and K. Sun, "Device-free gesture tracking using acoustic signals," in *Proceedings of the 22nd Annual International Conference on Mobile Computing and Networking*, ser. MobiCom '16. New York, NY, USA: Association for Computing Machinery, 2016, p. 82–94. [Online]. Available: https://doi.org/10.1145/2973750.2973764

[35] A. Famili, V. Slyusar, Y. H. Lee, and A. Stavrou, "Vehicular teamwork for better positioning," in *2023 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, 2023, pp. 3507–3513.

[36] A. B. Pizarro, J. P. Beltrán, M. Cominelli, F. Gringoli, and J. Widmer, "Accurate ubiquitous localization with off-the-shelf IEEE 802.11ac devices," in *Proceedings of the 19th Annual International Conference on Mobile Systems, Applications, and Services*, ser. MobiSys '21. New York, NY, USA: Association for Computing Machinery, 2021, p. 241–254. [Online]. Available: https://doi.org/10.1145/3458864.3468850

[37] G. Himona, A. Famili, A. Stavrou, V. Kovanis, and Y. Kominis, "Isochrons in tunable photonic oscillators and applications in precise positioning," in *Physics and Simulation of Optoelectronic Devices XXXI*, vol. 12415. SPIE, 2023, pp. 82–86.

[38] A. Famili and J.-M. J. Park, "ROLATIN: robust localization and tracking for indoor navigation of drones," in *2020 IEEE Wireless Communications and Networking Conference (WCNC) (IEEE WCNC 2020)*, Seoul, Korea (South), Apr. 2020.

[39] W. Mao, Z. Zhang, L. Qiu, J. He, Y. Cui, and S. Yun, "Indoor follow me drone," in *Proceedings of the 15th Annual International Conference on Mobile Systems, Applications, and Services*, ser. MobiSys '17. New York, NY, USA: ACM, 2017, pp. 345–358. [Online]. Available: http://doi.acm.org/10.1145/3081333.3081362

[40] D. F. Albuquerque, J. M. N. Vieira, S. I. Lopes, T. Aguilera, and F. J. Álvarez, "High energy ofdm pulse design algorithm for acoustic tof ranging and localization," in *2016 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, Oct 2016, pp. 1–6.

[41] A. Famili, A. Stavrou, H. Wang, and J.-M. J. Park, "RAIL: robust acoustic indoor localization for drones," in *2022 IEEE 95th Vehicular Technology Conference: (VTC2022-Spring)*, 2022, pp. 1–6.

[42] A. Famili, A. Tabrizian, T. Atalay, A. Stavrou, and P. Wei, "RAPID: reinforcement learning-aided femtocell placement for indoor drone localization," in *2024 33rd International Conference on Computer Communications and Networks (ICCCN)*, 2024, pp. 1–9.

[43] J. Roa, A. Jiménez, F. Seco, and J. Ealo, "Optimal placement of sensors for trilateration: Regular lattices vs meta-heuristic solutions," vol. 4739, 02 2007, pp. 780–787.

[44] T. Çavdar, F. B. Günay, N. Ebrahimpour, and M. T. Kakız, "An optimal anchor placement method for localization in large-scale wireless sensor networks." *Intelligent Automation & Soft Computing*, vol. 31, no. 2, 2022.

[45] N. Rajagopal, S. Chayapathy, B. Sinopoli, and A. Rowe, "Beacon placement for range-based indoor localization," in *2016 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, Oct 2016, pp. 1–8.

[46] N. Bulusu, J. Heidemann, and D. Estrin, "Adaptive beacon placement," in *Proceedings 21st International Conference on Distributed Computing Systems*, April 2001, pp. 489–498.

[47] J. B. Dai, Q. Fu, and N. K. S. Lee, "Beacon placement strategies in an ultrasonic positioning system," *IIE Transactions*, vol. 45, no. 5, pp. 477–493, 2013. [Online]. Available: https://doi.org/10.1080/0740817X.2011.649387

[48] P. Nazemzadeh, D. Fontanelli, and D. Macii, "Optimal placement of landmarks for indoor localization using sensors with a limited range," in *2016 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, Oct 2016, pp. 1–8.

[49] J. Si, R. Huang, Z. Li, H. Hu, Y. Jin, J. Cheng, and N. Al-Dhahir, "When spectrum sharing in cognitive networks meets deep reinforcement learning: Architecture, fundamentals, and challenges," *IEEE Network*, vol. 38, no. 1, pp. 187–195, 2024.

This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License. For more information, see https://creativecommons.org/licenses/by-nc-nd/4.0/