E-App: Adaptive mmWave Access Point Planning with Environmental Awareness in Wireless LANs

Yuchen Liu*, Mingzhe Chen†, Dongkuan Xu*, Zhaohui Yang‡, Shangqing Zhao§
*North Carolina State University, USA, †University of Miami, USA
‡Zhejiang University, China, §The University of Oklahoma, USA

2023 32nd International Conference on Computer Communications and Networks (ICCCN) | 979-8-3503-3618-4/23/$31.00 ©2023 IEEE | DOI: 10.1109/ICCCN58024.2023.10230133

***Abstract*—To enable ultra-high throughputs while addressing the potential blockage problem, maintaining an adaptive access point (AP) planning is critical to mmWave networking. By investigating the hidden interaction between the environment map and the placement of mmWave APs, we develop an adaptive AP planning (*E-app*) approach that can accurately sense the environment dynamics, reconstruct the obstacle map, and then predict the placements of mmWave APs adaptively. Specifically, our solution leverages mmWave radio itself to sniff the unacceptable performance degradation through sensing only a small fraction of observation points that are identified by a sparsity-aware analytical model, thereby accurately triggering a prediction module for AP positioning when necessary. Extensive evaluations show a very high prediction accuracy for our solution, which can provide around 25% improvement on user throughput performance in mmWave WLANs. This intelligent AP-planning framework well handles the environment dynamics that affect the average-case network performance, which is of utmost interest for network deployers because of its usage convenience and adaptivity.**

***Index Terms*—mmWave; AP planning; WLAN; prediction; environment-aware; reliabiliy.**

I. Introduction

Millimeter-wave (mmWave) communication has been heavily researched for academia and industry due to its huge available bandwidth, which has the potential to provide ultra high speed wireless communication with individual link rates over tens of gigabits per second. Because of this potential, mmWave is considered as a key enabling technology to support various bandwidth-intensive applications in recent years, e.g. virtual reality, online gaming, high-definition (HD) video streaming and holographic projection. Several standardization efforts, such as IEEE 802.11ad/ay, are focused on 60 GHz mmWave communications for wireless local-area networks (WLANs) [1], [2] with a goal of achieving multi-Gbps data rates to satisfy bandwidth-hungry applications.

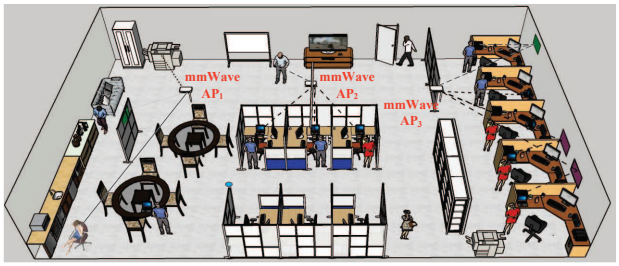
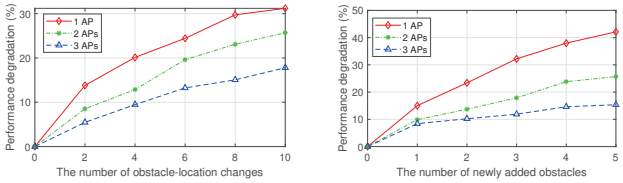


Fig. 1. A WLAN scenario with deployed mmWave access points (APs).

However, the achievable network performance of a mmWave WLAN is significantly sensitive to the environment dynamics such as blockages and user mobility. Prior researches on network planning [3]–[5] have found that the mmWave network performance is a sensitive function of access point (AP) placement and environmental characteristics, such as the obstacles' layout and locations of transceivers, and concluded that AP planning should be strategically determined to maximize the line-of-sight (LoS) coverage. However, considering the potential environment dynamics, i.e., as the obstacle map experiences a major change, e.g., some furniture is re-deployed or new furniture is brought in, the network performance could be possibly degraded a lot with the current multi-AP placement due to the location sensitivity of mmWave tranceivers, and this might require a position tuning of APs to maintain the network coverage and robustness, so as to enhance the overall network performance. To elucidate this situation, we did the performance study with environment dynamics in a 12m×8m×3m 60 GHz WLAN scenario, and evaluated the aggregate throughput performance over 20 user locations. As shown in Fig. 2, it is observed that as the obstacles layout (either locations or density) changes, the worst-case performance can be degraded 20%–40% even with multiple APs deployed. Another factor that may impact the AP planning is the popularity of user locations. This is driven by human behavior such that users may have unbalanced dwelling time at different locations, where wireless devices in frequently-visited areas (e.g., around conference or lunch tables) will generate or consume more data traffic. In the worst case, when all those popular hotpots fall in the unplanned shadowing region of deployed APs, the users' quality of service would be inevitably degraded since the large amount of data have to be transmitted under the worse channel conditions.



(a) The changes on locations. (b) The changes on density.

Fig. 2. Performance degradation vs. obstacles' locations and densities.

Based on above compound effects caused by environment dynamics, the initial AP deployment may not be sufficient

to maintain a desired network performance, and an adaptive multi-AP planning is required to update the AP placements when necessary, e.g., the obstacle layout or user-location popularity experiences a major change. However, the limited field of view of mmWave APs and sparse multipath diversity make it challenging to accurately sense the environment dynamics through mmWave radios. Therefore, recent works [6], [7] on deriving mmWave environment information have to rely on heavy infrastructure support, non-trivial human labor, or expensive software defined radios (SDR), which prevent their usage in practice. While in our preliminary research [3], we have derived the analytical model to obtain the shadowing regions in arbitrary environments, which makes it possible to map the physical environments with gathered LoS connectivity status at different locations. Thus, the work herein aims to investigate an environment-aware approach that leverages mmWave radios themselves to sense the environment dynamics, and reconstruct the obstacle map to guide an optimal multi-AP deployment adaptively. In general, through sensing the environment dynamics, the network will determine: 1) *whether* the network experiences an unacceptable performance degradation, which requires a multi-AP re-positioning; and 2) if so, *what* the new optimal multi-AP placements should be.

To this end, we propose an environment-aware AP planing (*E-app*) approach to guide an adaptive multi-AP placements due to the changes on obstacle layout and popularity of user locations. The re-positioning of APs could be performed through either human operation or automatic mobility techniques [8]–[10]. To be specific, in order to sense the environment changes and reconstruct the obstacle map, each AP first initializes a raw map that is divided into a number of small grids, and then a sparse sampling approach is used to identify the critical viewpoints that are highly correlated to the environment dynamics. Next, APs implicitly sense the LoS/non-LoS(NLoS) status for users at those selected viewpoints over time to determine whether or not the network experiences an unacceptable performance degradation. In parallel, a deep neural network based prediction model is trained with the synthetic generated dataset to infer the full obstacle map, which in turn helps optimize and tune the multi-AP placements when necessary. Extensive evaluations show that our *E-app* can achieve fairly high accuracy to predict the optimal AP placements under environment dynamics, and provide around 25% improvement on user throughput performance in mmWave WLANs.

The main contributions of this work are as follows.

- We propose the design of an adaptive AP planning scheme (*E-app*) that includes the sparse sensing strategy and the optimal AP-placement prediction to deal with the performance degradation brought by environment dynamics in mmWave WLANs. This approach can synergistically determine *when* and *how* to optimally place or re-deploy the APs, which provides a basis for the development of BS/AP mobility technique in the wireless network with *automatic location discovery* capability.
- We develop for the first time a method to quantify reliability of the location points in mmWave WLANs, which is used to identify only a few sensitive observation points that are highly correlated with the environment dynamics. The resultant sparsity-aware scheme is able to significantly reduce the environmental sensing overhead used for data collection and training process.
- We develop the systematic solution of *E-app* and perform extensive evaluations, where the results show the highly accurate prediction to AP planning under environment dynamics and significant improvement on the overall network performance in mmWave WLANs. This demonstrates that mmWave AP placement can be accurately predicted through the use of detailed environment information and the awareness of a few critical observation locations.

II. Related Work

Prior works that have addressed the problem considered herein, i.e. multi-AP planning in indoor mmWave settings are [3], [6], [11]–[14]. Of these, [11] investigated the impact of base station deployment on LoS probability in 5G indoor scenarios, [3] proposed the LoS-optimal AP planning schemes in mmWave WLANs, and other AP placement schemes were studied in [12]–[14] to evaluate random, edge-based, or evenly-spaced multi-AP deployments. Each of the above planning schemes is a heuristic or LoS-optimal scheme, however, they all failed to overcome the performance degradation brough by potential environment dynamics since the planning was only performed for once without considering the re-deployment of APs when necessary, which is the main subject of our work herein.

Specifically, a more relevant work about 60 GHz AP planning was presented in [6], where sensed environment information is fed into a ray tracer to predict link performance and then guide the AP deployment. There are several key differences between the approach of [6] and our work. First, [6] relies heavily on software-defined radios, where a dedicated and specialized hardware platform is used to provide a detailed map of the propagation environment in order to guide the AP deployment. By contrast, our approach only needs to use mmWave radios and LoS information. Second, the approach in [6] does not attempt to capture environment dynamics, and in case such obstacle change their locations, the SDR infrastructure has to be deployed to accommodate the changes, i.e., moving transceiver to multiple locations for measurement. While our solution can well handle the environment dynamics automatically with implicitly sensing a few observation points by APs, which provides good performance over a wide range of obstacle environments and guides the optimal AP (re)deployment when necessary.

For the prior works on sensing the environment dynamics, state-of-the-art radio-based simultaneous localization and mapping [15], [16] can only achieve localization accuracy of around 5 meters, far from enough to predict the spatial performance of a mmWave wireless network in a typical indoor scenario. Some works [17], [18] adopted multiple mmWave radars to explicitly scan the environment by continuously

moving the radar in front of the obstacle's body to identify its shape and reflectivity. In contrast, our work leverages the collected LoS channel information from 60 GHz Wi-Fi APs, so as to reconstruct the entire environments with limited sampling client spots. Another benefit of our work is that we can sense the environment dynamics irrespective of the number of available APs, i.e., even a single AP can also discover the environment details and thereby recovering an obstacle map, which in turn helps optimize the planning adaptively. Also, other works [7], [19]–[22] proposed to sense the environment through assistants of mobile robot, radar, ray-tracing engine, or laser localization technology and, however, these techniques require additional deployment of sensors, which increases the implementation and hardware complexity. Our work stands apart from such side-information aided schemes since our approach does not require any additional infrastructure support, which only leverages mmWave radios themselves as sensors to "sniff" the environment and guide an optimal AP planning with the better usage convenience and adaptivity.

III. E-app: An Overview

E-app is the first holistic scheme for optimizing and tuning AP deployments over mmWave WLANs, taking into account the underlying environment dynamics. The major challenges in deriving this solution are: 1) A small change on obstacle map may have a dramatic impact on AP deployment to maximize network performance, thus the knowledge of the environment dynamics is essential to guide and tune the AP planning adaptively. However, it is analytically intractable to determine the optimal AP placements as environment changes; 2) Although machine learning techniques could be used to make predictions on AP planning, it is challenging to collect a sufficient volume of training data in real environment covering a complex range of network scenarios; and 3) in practice, the collected sensing information from APs is either incomplete or redundant, which makes it infeasible to reconstruct the entire obstacle map, thus posing the challenge on tuning and redeploy the APs when necessary.

Fig. 3 gives an overview of our *E-app* framework that addresses the above challenges, which includes the process of online inference and offline training. In the online inference stream, a 3D weighted shadowing-elimination searching (wSES) algorithm is first proposed to efficiently determine the optimal AP placements, and a set of shadowing-region (SR) maps are also derived that contains the LoS/NLoS information at arbitrary locations of the scenario. Then, a sparse sampling approach is investigated to identify only a few sensitive observation points (OPs) in SR maps that are sufficient to reflect the environment dynamics. Next, APs keep sensing the LoS/NLoS status of the users located at those OPs over time, and the LoS coverage rate (LCR) among the OPs will be constantly checked. Once LCR is lower than expected, i.e., the environment map experiences a major change, a deep neural network (DNN) will be run to predict the new optimal AP placements. The DNN model is derived from the offline training stream as shown in Fig. 3, which is in parallel with the online inference phase. Specifically, we propose a data generation framework to efficiently generate a large amount of synthetic data by reusing the wSES algorithm, which are then used to train the DNN model for predictions to new optimal AP placements with respect to sensed dynamics. Lastly, the redeployment of APs can be performed through either simply human operation or automatic mobility techniques as in [8]. It is worth noting that the proposed *E-app* also offers the potential solution for the *location discovery* problem of BS/AP, UAV, or robot mobility technology used in future wireless systems, e.g., proactively moving an AP to the optimal locations that maximize the network coverage adaptively, or developing the optimal trajectory of UAV-based BSs to offload services to users within a region of interest. We will leave these promising directions as the future work, and in what follows, we discuss the details of technical components in our *E-app* framework.

IV. Space Gridding: Weighted Shadowing Area Search

Considering the sharp differences between LoS and NLoS performance in mmWave wireless network [23], we use geometric analysis to identify the shadowed regions in an area that correspond to definite LoS/NLoS cases, as shown in the first step of Fig. 3. To capture the environment dynamics and user location preference, we propose a weighted shadowing-elimination search (wSES) approach that jointly considers obstacle effects and client-location popularity to efficiently determine the LoS and NLoS areas and the optimal placement of APs in a given scenario. Algorithm 1 summarizes the steps of wSES algorithm, involving the following two procedures:

1) Space gridding: The network environment space S is divided into N_c equal sized *cubes* c_i with size length of l_c, where $\bigcup_{i=1}^{N_c} c_i = S$ and $\bigcap_{i=1}^{N_c} c_i = \emptyset$. The center of each cube can be considered as a candidate observation point. From the 2D perspective, the network environment space S is actually decomposed as $\lceil r_h/l_c \rceil$ planes, where each of them consists of N_g equal sized grids with the gridding length of l_c, where $N_g = (r_l \cdot r_w)/l_c$. In practice, we consider that the height range of a client devices is $H = [H_{min}, H_{max}]$, so the space is divided into $\lceil H_{max} - H_{min}/l_c \rceil$ 2D planes, and each plane has N_g non-overlapped grids.

2) Weighted shadowing elimination: Based on the knowledge about the obstacle information Obs and popularity of user locations P_w[1], Algorithm 1 first considers all grids in a 2D plane of room at the specific height (Lines 1-2), and then the weighted shadowing volume wSV with regard to each AP position is calculated (Lines 4-5). Specifically, we divide the 3-D space into several 2-D planes at different heights and therefore, the obstacles' heights need to be adjusted accordingly due to the change of each height base h_i (Lines 13-14). Next, for every obstacle i at height base h_i, a grid-based searching method from our prior work [3] is

[1]In practice, the objects' locations, sizes, and material types could be obtained in a variety of ways, e.g., through camera-based sensing [24], and P_w can be investigated by recording user's frequently visited locations (FL) within the room over time [25].

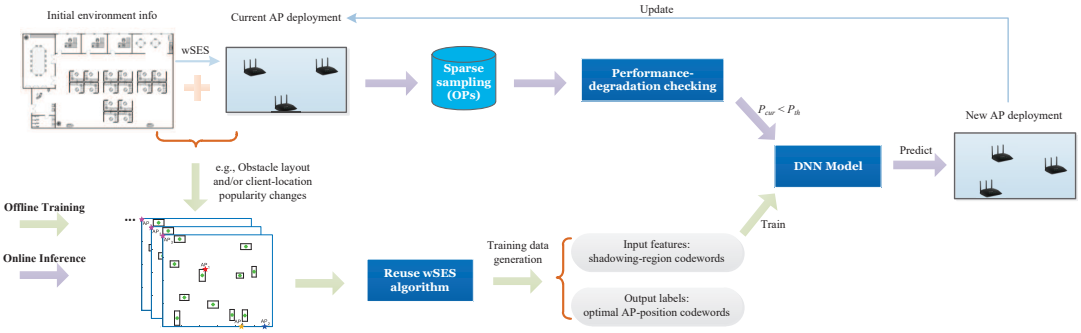


Fig. 3. Schematic of proposed *E-app* framework.

conducted to get shadowed-grid set SG_j, which includes all grids shadowed by the obstacle (Lines 16-17). Next, we start to check whether the grid in SG_j has already been put in SG_{all} (Lines 18-19), and if not, this new shadowed grid will be added in SG_{all} (Line 20). This step is used to eliminate the overlapping shadowing area caused by different obstacles. After traversing all obstacles at all height bases, the weighted shadowing volume is obtained as:

$$wSV = \sum_i P_{w_i} \cdot l_c^3 \ (i \in SG_{all}), \tag{1}$$

and the first AP is optimally placed in the position with the minimum wSV (Lines 6-7). Before starting to find the next AP's position, the grid set G is updated as the shadowed-grid set of the first AP (Line 8), which means that the second AP will be placed at the position that eliminates the most remaining wSV of the first AP. Then, the third AP is placed to minimize the remaining shadowing volumes given the first two APs positions. This process continues until all N APs' positions are derived. In [3], this SES-based approach has been validated to achieve the near-optimal results for AP placement.

As a side note, the wSES algorithm can also output the shadowed grids G over the entire 3D space, which is used to map the physical environment from a camera's eye to a radio "environment" from APs' eyes (i.e., a set of shadowing-region map) as shown in Fig. 4.

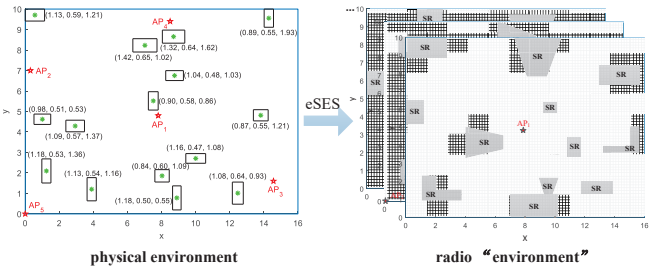


Fig. 4. From physical environment to radio "environment".

V. Sparse Sampling: Critical Viewpoint Identification

To sense the environment dynamics through LoS identification technique [26] from APs' eyes, we propose a sparse sampling approach to identify the critical viewpoints on the radio environment map. The underlying principle suggests the potential to collect a smaller-sized network information by *pruning* useless information while resulting in equally optimal solutions for the original problem. The sensing overhead to detect environment changes will be significantly reduced by imposing sparsity awareness, as we can sense the LoS/NLoS information from only a few sensitive observation points that obviously reflect the environment dynamics, which further helps *reconstruct* the new shadowing-region map.

Definition 1. Client locations p_i are defined as observation points (OP) only when their client-dwelling probability exceeds a threshold ϵ_p (e.g., 0.05), i.e.,

$$OP = \{p_i \in G | P_{w_i} > \varepsilon_p, \ h_{p_i} \in H\} \tag{2}$$

Definition 2. An OP has the *state transition* if it is under one of the following situations: 1) the OP had at least one LoS APs, but now there is no LoS APs to it after any environment dynamics; 2) the OP was under a NLoS condition, but now there exists at least one LoS AP to it.

Based on our observation, not all OPs are sensitive to environment changes, which means that some of them (referred to as *inert points*) seldom or never experience the state transition in terms of the current AP deployment. This motivates us to identify the *volatile points* (VPs) as critical viewpoints for sensing, which are actually sufficient to capture the potential changes on an obstacle map. This sampling process can significantly reduce the computational overhead of online inference process as shown in Fig. 3. In what follows, we first derive the new metric to quantify the "volatility" of all OPs, and then develop the algorithms to identify these VPs.

A. Reliability Score

Starting from the single-AP case, we first use geometric analysis to quantify the reliability of the link between a client location and an AP. Based on the random shape theory from [27], obstacles are assumed to form a Boolean scheme of rectangles, in which the centers of the rectangles fall within the room and form a homogeneous Poisson point process (PPP) C of density λ, where λ is the mean number of obstacles in an unit area. The width, W, and length, L, are assumed to be i.i.d. distributed and follow normal distributions, i.e. $W \sim \mathcal{N}(\mu_w \ {\sigma_w}^2)$ and $L \sim \mathcal{N}(\mu_l, {\sigma_l}^2)$. The orientation distribution Θ of every obstacle is uniform over $[0, \pi]$. In this way, a

Algorithm 1: wSES: weighted Shadowing-Elimination Search

Input: Obs, l_c, prm, N, P_w (popularity weights), H_{min}, H_{max}
Output: P_{ap}

for *each height* ($h_i = H_{min} + i * l_c$) & ($h_i <= H_{max}$) **do**
  $G(i,:)$=[all grids in a 2D plane at height h_i]; ▷ init G
for *each AP* i *from 1 to* N **do**
  for *each pos of AP* i **do**
    [wSV_{u_i}, SG_i]=FindSV(Obs, pos, G, l_c, prm, P_w);
  AP_i=arg min{wSV_{u_i}};
  P_{ap}.add(AP_i);
  G=[grids in SG_i]; ▷ update G
  if $G = \emptyset$ **then break**;
return P_{ap};
$Function$: FindSV(Obs, pos, G, l_c, prm, P_w)
Init SG_{all}; ▷ init the shadowed-grid set
for *each height* ($h_i = H_{min} + i * l_c$) & ($h_i <= H_{max}$) **do**
  Obs.height = max{Obs.height - $h_i, 0$}; ▷ change the height base
  pos.height = pos.height - h_i; ▷ change the height base
  for *each obstacle* $j \in Obs$ **do**
    [SA_j, SG_j] = GSSFunction($Obs(j)$, pos, $G(i,:)$, prm); ▷ Tech Report
    for *each grid* $g \in SG_j$ **do**
      if ($g \notin SG_{all}$) **then**
        SG_{all}.add(g);
wSV_u=$\sum_i P_{w_i} \cdot l_c^3$ ($i \in SG_{all}$); ▷ weighted shadowing volume
return wSV_u, SG_{all};

random obstacle is characterized by the quadruple $\{c, w, l, \theta\}$, which is generated by sampling the distributions C, W, L, Θ.

Based on the blockage-area calculation in a 2D plane, we derive the expectation of total number of random obstacles blocking the link is $\frac{2d_i \cdot (\mu_w + \mu_l)}{\pi} + \mu_w \cdot \mu_l$, where d_i is the horizontal distance between AP and client. Then, we introduce the height effects of obstacles and extend the model to 3 dimensions. Based on the height modeling from [25] and assuming that the obstacle's height h_o follows the uniform distribution $H \sim U(a_o, b_o)$, we derive the conditional probability that an obstacle blocks the connected link is:

$$\varepsilon = 1 - \int_0^1 \int_0^{y \cdot H_c + (1-y)H_A} f_H(h)\ dh\ dy\ = 1 - \frac{1}{b_o - a_o} \int_0^1 [\max\{\min\{b_o, yH_c + (1-y)H_A\}, a_o\} - a_o]\ dy = \frac{b_o + a_o - 2H_c}{2(H_A - H_c)}, \tag{3}$$

where H_A and H_c are the height of AP and client location ($H_A > H_c$), respectively. Next, we arrive at the probability that the LoS link exists with multiple random obstacles, which is also referred to as *reliability score* (RS):

$$RS_1 \overset{(a)}{=} e^{\frac{\lambda}{2\pi(H_A - H_c)} \cdot (2H_c - b_o - a_o)(2d_i\mu_l + 2d_i\mu_w + \pi\mu_l\mu_w)} \tag{4}$$

where (a) follows the basic property of homogeneous Poisson distribution with density λ.

While extending the derivation of RS to multi-AP cases is not straightforward, we first introduce the RS derivation for 2-AP case. In a similar way, we first derive the blockage area from an individual link, and then compute the overlapping blockage area of two links, finally incorporating the height effects in Eq. 3, such that we arrive at the close-form expression of RS score in 2-AP case, i.e., the probability that at least one of the LoS links is connected with multiple random obstacles:

$$RS_2 \overset{(b)}{=} \sum_{x=i,j} RS_1(d_x) + e^{\frac{-2\lambda\varepsilon(d_i + d_j)(\mu_l + \mu_w) + 2\pi\mu_l\mu_w}{\pi}} \cdot e^{\frac{\lambda\varepsilon({\mu_w}^2 + {\sigma_w}^2)}{\pi} \cdot [\frac{1}{4}(\pi - \alpha) \cdot \cot\gamma - \frac{\sin(2\alpha - \gamma)}{8\sin\gamma} - \frac{1}{4}\cos^2\gamma + \frac{1}{8}]} \cdot e^{\frac{\lambda\varepsilon({\mu_l}^2 + {\sigma_l}^2)}{\pi} \cdot [\frac{1}{4}(\pi - \gamma) \cdot \cot\gamma + \frac{1}{4}\cos^2\gamma]} \cdot e^{\frac{\lambda\varepsilon\mu_w\mu_l}{2\pi}(\pi + \gamma)} \tag{5}$$

where (b) is true because of the inclusion-exclusion principle and the property of homogeneous Poisson distribution. In Eq. 5, α equals to $\max\{0, \gamma - \pi/2\}$, and γ is the intersected angle of two links.

It is not trivial to extend the RS analysis to higher-dimensional AP case, but here we adopt a simplified geometric model from [28], i.e., the line Boolean model, which assumes that all the blockage elements are in the form of lines for tractability. Based on this model, the RS derivation can be easily extended to N-AP cases ($N \geq 3$).

In 3-AP case, based on the line Boolean model and Eq. 3, the probability that no blockages lie in a link l_i of distance d_i between AP_i and client is derived as $e^{-\lambda S_{l_i}}$, where $S_{l_i} = \frac{2\lambda\epsilon\mu_l d_i}{\pi}$. Then, considering any two links (l_i and l_j) between the client and APs, the overlapped blockage area S_{l_i,l_j} can be derived through the geometric analysis as follow:

$$\begin{aligned} S_{l_i,l_j} &= \int_l \int_\gamma^{2\pi} \frac{l^2 \sin\theta \cdot \sin(\theta - \gamma)}{2\sin\gamma} \cdot [1 - (1 - \min\{1, \frac{d_i \sin\gamma}{l\sin\theta}, \\ &\quad \frac{d_j \sin\gamma}{l\sin(\theta - \gamma)}\})^2] f_L(l) f_\Theta(\theta)\ dl\ d\theta \\ &= \tfrac{2}{\pi}\mu_l(d_i + d_j) - \tfrac{1}{4\pi}({\mu_l}^2 + {\sigma_l}^2) \cdot [(\pi - \gamma_{i,j}) \\ &\quad \cdot \cot\gamma_{i,j} + 1] \end{aligned} \tag{6}$$

where $d_{i(j)}$ is the distance between $\text{AP}_i(j)$ and client, and $\gamma_{i,j}$ is the intersection angle of l_i and l_j. Moving forward, we derive the overlapped blockage area with all three links in a similar way:

$$\begin{aligned} S_{l_i,l_j,l_k} &= \tfrac{2}{\pi}\mu_l \sum_{x \in i,j,k} d_x - \tfrac{1}{4\pi}({\mu_l}^2 + {\sigma_l}^2)[(\pi - \Gamma_{\min_1}) \\ &\quad \cdot \cot\Gamma_{\min_1} + (\pi - \Gamma_{\min_2}) \cdot \cot\Gamma_{\min_2} + 2] \end{aligned} \tag{7}$$

where $\Gamma_{\min_1}$ and $\Gamma_{\min_2}$ are the first two minimal intersected angles among the three links. Finally, we arrive at the RS of three-AP case as:

$$RS_3 \overset{(b)}{=} \sum_{x\in i,j,k} e^{-\frac{2\varepsilon\lambda\mu_l d_x}{\pi}} - \sum_{x,y\in i,j,k;x\neq y} e^{-\lambda\varepsilon S_{l_x,l_y}} + e^{-\lambda\varepsilon S_{l_i,l_j,l_k}} \tag{8}$$

Without the loss of generality, we derive the RS for multi-AP cases (more than three APs) as follow:

$$\begin{aligned} RS_n = P(\bigcup_{i=1}^{n} S_{l_i}) &= \sum_{B\subseteq A_s} (-1)^{|B|-1} \cdot P(\bigcap_{s\in B} s) \\ &\overset{(c)}{\approx} \sum_{i\leq n} P(S_{l_i}) - \sum_{i,j\leq n;i\neq j} P(S_{l_i,l_j}) \\ &\quad + \sum_{i,j,k\leq n;i\neq j\neq k} P(S_{l_i,l_j,l_k}) \\ &\quad + o(\sum_{B\subseteq A_s\setminus(S_{pre})} (-1)^{|B|-1} \cdot P(\bigcap_{s\in B} s)) \\ &\overset{(d)}{=} \sum_{i} e^{-\frac{2\varepsilon\lambda\mu_l d_i}{\pi}} - \sum_{i,j\leq n;i\neq j} e^{-\lambda\varepsilon S_{l_i,l_j}} \\ &\quad + \sum_{i,j,k\leq n;i\neq j\neq k} e^{-\lambda\varepsilon S_{l_i,l_j,l_k}} \quad (n>2) \end{aligned} \tag{9}$$

where (c) is true because of the multi-set inclusion-exclusion principle and the fact that APs are distributed dispersively in the scenario, which always leads to the negligible overlapped blockage areas of more than 3 links, and (d) is based on the homogeneous Poisson distribution and Eq. (6)-(8).

B. Impact factor and algorithm design

Based on the derived RS, we define a new metric to quantify the "volatility" (i.e., unreliability) of every possible observation point. Intuitively, this impact factor (IF) should reflect 1) how sensitive the OP is to the environment dynamics; and 2) how popular the OP is (i.e., whether the client location is frequently visited or not). To this end, by incorporating the popularity weight (P_w), we arrive at IF of each OP i as:

$$IF_i = P_{w_i} \cdot (1 - RS_i). \tag{10}$$

In specific, IF is higher when the OP has i) a larger popularity weight, and/or ii) a lower reliability score, i.e. more susceptible to blockage effects. Fig. 5 shows IF distributions of different client locations in a network scenario[2], and it is observed that the IF distribution is highly dependent on the density and location of APs. Therefore, based on the current AP deployment, it has the potential to identify the VPs among all OPs in the network scenario.

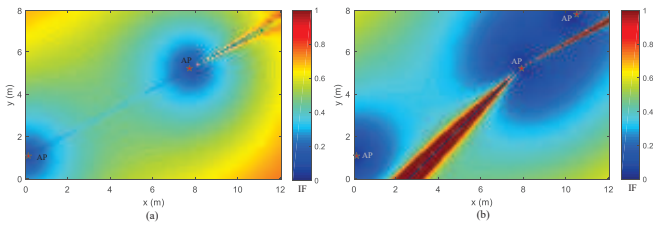


Fig. 5. IF distribution of OPs in a network scenario. (a) 2-AP case; (b) 3-AP case.

[2]Here we set $\mu_l = 1.08$, $\sigma_l = 0.18$, $\mu_w = 0.56$, $\sigma_w = 0.08$, $a_o = 1.5$, $b_o = 2.5$, $H_c = 1.5$, $H_A = 3$, and P_{w_i} is equally distributed.

Algorithm 2 shows the approach to identify the VPs. Considering all possible cubes (i.e., all available OPs), we first compute their IF factors (Lines 1-4), and rank these cubes v in the descending order of IF (Line 5). Then we re-scale IF with the min-max normalization (Lines 6-7). Next, starting from the viewpoint v_i with the largest IF, we put the viewpoint v_i into the VP set only if the distances to any other existing VPs are greater than their robustness coherence distances (RCD) (Lines 8-13). We define RCD as the ratio of reference distance δ and the normalized IF_i, where the viewpoint with larger IF will have the smaller RCD, such that more OPs around it are likely to be selected as VPs, and this is consistent with the fact that mmWave spatial channel profiles at nearby locations are *highly-correlated*. Thus, the OPs having better reliability (i.e., the smaller IF) will be given a larger RCD since the nearby viewpoints within its RCD are expected to show the similar robustness to the environment changes. Note that δ is a tuning parameter to determine the percentage of sampling OPs that will be selected as VPs.

Algorithm 2: Identify the critical viewpoints

Input: $apPos$, δ, P_w, H, l_c
Output: VP

1 G = GetCube(H, l_c);
2 **for** *each cube center* $v_i \in G$ **do**
3 $\quad IF_i = P_{w_i} \cdot (1 - RS_i)$;
4 $\quad Pair$.add(v_i, IF_i);
5 $Pair$.order($\downarrow IF$);
6 **for** *each* $IF_i \in Pair$ **do**
7 $\quad IF_i = \frac{IF_i - \min(IF)}{\max(IF) - \min(IF)}$; ▷ min-max normalization
8 **for** *viewpoint from* v_1 *to* v_n *in* $Pair$ **do**
9 $\quad VP$.add(v_i);
10 $\quad$ **for** *each viewpoint* $v_j \in VP$ **do**
11 $\quad\quad$ **if** $||v_i - v_j|| \leq \frac{\delta}{IF_j}$ **then**
12 $\quad\quad\quad VP$.rmv(v_i);
13 $\quad\quad\quad$ **break**;
14 **return** VP;

To show the performance of our Algorithm 2, we evaluate the VP coverage rate (VCR) vs. the percentage of sampling OPs (see Fig. 6), where VCR indicates the percentage of OPs having status transitions due to environment changes covered in the set of identified VPs, and 100% VCR means that all OPs experiencing the state transitions have been included in the VP set. Each data point in Fig. 6 is the averaged result of over totally 1000 dynamic cases over 5 different scenarios[3]. From the figure, we can see that our algorithm can accurately identify the VPs among all OPs, and only 30%–40% OPs are sufficient to reach almost 100% VCR with different number of APs. For example, in a scenario with 200 OPs, we can identify only 60 VPs used to capture the environment dynamics, which significantly reduces the overhead of the following information collection and data training processes.

[3]For simplicity, we assume the equal P_w in each OP in this evaluation.

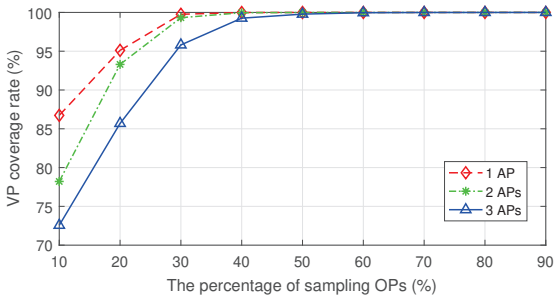


Fig. 6. VCR vs. the percentage of sampling OPs.

VI. E-app: Environment-Aware AP Planning

Based on the above approaches to determine the optimal AP placements (in Sec. IV) and to identify the critical viewpoints for sensing environment dynamics (in Sec. V), here we introduce a data-driven approach to predict the AP placements adaptively, and the overall systematic solution of *E-app* will be shown in detail.

A. Optimal AP placement prediction

As shown in Fig. 3, one of the key components of our proposed *E-app* framework is a deep learning based model to predict the optimal multi-AP placements adaptively, which involves the dataset generation and learning model design, discussed next.

First, as shown in the bottom half of Fig. 3, Algorithm 1 is performed to collect a large amount of environment maps and corresponding optimal AP deployments, where each data instance is generated with a random-obstacle distribution to simulate the environment dynamics based on the initial scenario configuration, i.e., the density and dimensions of obstacles constantly vary in the network scenario. Next, we train a deep learning based approach to predict the optimal AP placements under those environment changes. The proposed learning-based approach takes into account multiple network state information as input, where the time complexity will be constant for an offline trained model. Specifically, the problem to predict the optimal positioning of APs is represented and solved in a supervised fashion. Thus, the underlying relationship between input and output is actually a skewed representation of the obstacle map, and we utilize deep neural networks (DNNs) as a recipe for parametric function approximation to learn this latent structure.

1) Input feature: The input data is present in the format of LoS connectivity matrix (LCM) between OPs and some AP_i, i.e., $LoS_{(ac,t)}$. Specifically, we incorporate the error model of LoS estimation and localization based on the prediction cumulative distribution function (CDF) presented in [26] and in [29], respectively. Given the network scenario with N APs, room's length r_l and width r_w, and the grid size of g_l, each $LoS_{(ac,t)}$ matrix has the size of $(r_l \cdot r_w)/g_l$. Therefore, the input feature vector X is obtained by concatenating the feature vectors of all APs into a single vector of size $N \cdot (r_l \cdot r_w)/g_l$.

2) Output label: The labels (i.e., ground truth) for training are present in the format of $Pos_{(N,t)}$, i.e., the multi-AP positioning matrix for all APs including $(r_l \cdot r_w)/g_l$ locations in a 2D plane as the APs are always mounted on the ceiling to maximize the LoS coverage. Specifically, in the data-generation step, we have obtained the two dimensional cartesian coordinates of each AP's position. Then, we perform a pre-processing procedure to transform these coordinates into a $[0,1]^{(r_l \cdot r_w)/g_l}$ vector $Pos_{(N,t)}$, which contains N ones and $N - (r_l \cdot r_w)/g_l$ zeros, and the grid index with "1" is corresponding to the two dimensional cartesian coordinates of APs. In this way, the network outputs $\hat{Y} \in [0,1]^{(r_l \cdot r_w)/g_l}$, which is a $(r_l \cdot r_w)/g_l$ sized probability vector representing the probability of optimal positions of N APs.

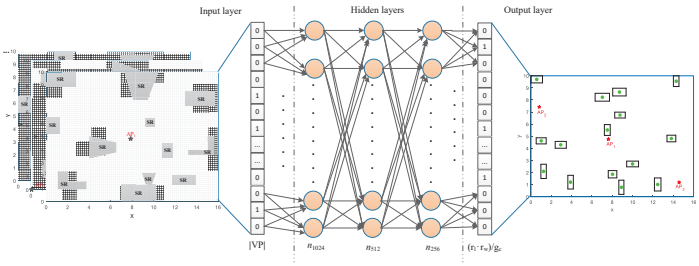


Fig. 7. Neuron network architecture.

3) Network configuration: Based on the input features and output labels, we design a DNN with a number of hidden layers and neurons configured to work across different network scenarios. The flattened input feature vector of size $N \cdot (r_l \cdot r_w)/g_l$ is fed to a fully connected network as shown in Fig. 7 with three hidden layers. The l^{th} hidden layer has a total of n_{H_l} neurons. The k^{th} neuron in $(l-1)^{th}$ layer is connected to j^{th} neuron in l^{th} layer with a weight of w_{jk}^l. b_j^l represents the bias of the j^{th} neuron in the l^{th} layer. The activation of the j^{th} neuron in the l^{th} layer, i.e. a_j^l, is calculated through the forward propagation rule as below,

$$a_j^l = f(\sum_k w_{jk}^l a_k^{l-1} + b_j^l), \tag{11}$$

where f applies the non-linearity in the model using the *tanh* activation function. Then, we use a softmax layer before the output layer to transform the output logits to the probability vectors. The model is trained through the backpropagation rule, using weighted cross-entropy loss as:

$$H_y(p) = \sum_i^P -(y_i \log(p_i) * w + (1-y_i)\log(1-p_i)), \tag{12}$$

where p represents the softmax probability of output logits, and w is calculated as the ratio of number of shadowed grids (N_{sg}) vs. non-shadowed grids (N_{ng}) according to the training data. As the ratio of N_{ng} to N_{sg} in the data samples is imbalanced, the weighted cross entropy loss with weight w can balance the loss function to avoid any local minima.

With the available training data bank, $DB = \{(X_1,Y_1),(X_2,Y_2),\ldots(X_N,Y_N)\}$, of N samples, the loss function is minimized using adaptive moment estimation optimizer, where a batch of B training samples is randomly selected out of N training samples, and the weights and biases are updated through the backpropagation rule. A fraction of

the gradient in the previous iteration is retained with the "coefficient of momentum". At each learning iteration, the learning rate is decreased over time to optimize performance and to increase the convergence rate of the algorithm. While training, we also augment the training set by a random permutation over the sequence order of the APs' LCMs in the input features. This not only increases the training set size but also improves the convergence of gradient descent by avoiding any OP order-based local minima, and the random permutations prevent the DNN architecture from extracting features based on the APs ordering.

Note that the machine learning algorithm in our *E-app* is not new, however, its application to this specific planning problem, and the overall solution design is novel. Briefly, the novelty of the overall learning model lies in 1) the intuition behind the learning solution, e.g., the underlying obstacle map is *discoverable* through the LoS connectivity information between APs and a few observation points; 2) the methodologies to make the model *generalizable* through augmented training; and 3) as we will show later in Sec. VII, a single network architecture with all the hyper-parameters capable of learning in different network scenarios.

B. Systematic solution

Algorithm 3 summarizes the overall *E-app* solution executed by the network controller. First, based on the initial scenario configuration, the APs are optimally placed using the wSES algorithm (Line 1). Then, the critical viewpoints VP will be identified and transmitted to APs for sensing (Lines 2-3). At start with the current AP deployment, all APs collect LoS connectivity information (LI) among those VPs, and then compute the weighted shadowing factor (WS_{org}) for record (Lines 4-5). In the meantime, a bunch of training data are generated with respect to the current AP deployment for training the DNN-based AP positioning prediction model (Line 6) based on the sythetically generated dataset. During each time period, network controller collects the updated LI from APs and computes the current weighted shadowing factors WS_{cur} (Lines 7-9). Once the network experiences a non-negligible performance degradation due to the environment changes, the LI from each AP will be reformed as the input features to the trained DNN model, which then predicts the corresponding optimal AP deployment under current situation (Lines 9-15). Once the new AP placements are made, the new round of computation, sensing, and data training process with regard to the current AP locations will be repeated in the same way.

Algorithm 3: Workflow: *E-app* solution

Input: $N_{ap}, T, t_s, r_l, r_w, l_c, prm_o, P_w, H$
Output: $apPos_{new}$

1 $apPos_{cur}$ = wSES(r_l, r_w, l_c, prm_o, P_w, H); ▷ Sec. IV
2 VP = IdentifyVPs($apPos_{cur}$, P_w); ▷ Sec. V
3 Inform APs with VPs;
4 Collect initial LIs from APs;
5 $WS_{org} = \sum_k P_{w_k} \cdot \mathbb{I}_k(\cap_i LI_{init})$; ▷ $\mathbb{I}_k(\cdot)$ is the indicator function, which is equal to 1 if the value of CP(k) is 1 in $\cap_i LI_{init}$.
6 Train a DNN model with the training data[4]; ▷ Sec. VI
7 **while** *each time step* t_s *at* t_j **do**
8 Collect LIs from APs;
9 $WS_{cur} = \sum_k P_{w_k} \cdot \mathbb{I}_k(\cap_i LT_i)$; ▷ only consider the critical grids
10 **if** $(WS_{org} - WS_{cur})/WS_{org} > T$ **then**
11 **break**; ▷ stop detection phase, update planning
12 ▷****Update AP planning****◁
13 LI_{re} = Reform LIs: ($P_{w_i} \rightarrow 0$, $0 \rightarrow 1$); ▷ consistent with the form of input in the prediction model
14 $input$ = concatenate(LI_{re_i} of each AP_i);
15 $apPos_{new}$ = Prediction($input$); ▷ predict the AP positions
16 **return** $apPos_{new}$;

VII. Evaluation Results

In this section, we evaluate the performance of our *E-app* scheme to optimize and reconfigure the AP placements in mmWave WLANs.

A. Network Settings

We first randomly generate various WLAN scenarios with the following features: 1) the lengths, widths, and heights of rectangular room follow uniform distributions $\text{L}_r \sim \mathcal{U}$(10.0, 20.0), $\text{W}_r \sim \mathcal{U}$(5.0, 10.0), and $\text{H}_r \sim \mathcal{U}$(2.4, 4.5); 2) Objects deployed in each scenario are modeled as cuboids and placed on the floor, where the center of each obstacle follows a Poisson point process with a specific density $\lambda \sim \mathcal{U}$(0.04, 0.3), the widths, and lengths follow the truncated normal distributions W$\sim \mathcal{TN}$(0.56, σ_w, 0.25, 1.25) and L$\sim \mathcal{TN}$(1.08, σ_l, 0.5, 1.75), where $\sigma_w \sim \mathcal{U}$(0.01, 0.38) and $\sigma_l \sim \mathcal{U}$(0.08, 0.58). Their heights and orientations follow uniform distributions $\Theta \sim \mathcal{U}(0, \pi)$ and H $\sim \mathcal{U}$(0.3, 2.3)); 3) each scenario case includes some client devices, where are viewed as a random point, and its height follows the uniform distribution $\mathcal{U}$(0.1, 2.0). Also, a number of hotspot locations with different popularity weights are generated in each scenario. These parameters are derived by using a real-life office/lab environment as a guiding example, and all length units of parameters are in meters. Then, by running wSES algorithm, the APs are initially deployed in the specific positions such that the network achieves maximum LoS-coverage operation. Next, the environment dynamics randomly occur in the network scenario, including the changes on density, locations of the obstacles and the user-location popularity.

In our DNN model as shown in Fig. 7, three hidden layers in the model have 1024, 512 and 256 neurons, respectively. We use a default batch size of 1024 (or more) except for the cases where the total training sample size is smaller than 1024. The learning rate is initialized as 0.01, and decreased with a factor of 0.9 every 2000 steps. We split the available data into two sets: 1) the training set comprises of 80% of the data and is used to learn the network weights and bias; and 2) the remaining 20% set is used for testing the predicted

[4]Note that this data generation time is not a serious issue, because it only has to be done once to generate the model and then it can be used as many times as needed for different scenario settings.

results. For each configuration, we ensure the evaluation correctness through random permutation tests. Note that this is a complicated classification problem as predicting some AP should be optimally placed in a grid-level position and hence, a random classifier will have a low accuracy of $(0.5)^{N_g}$. Additionally, we randomly permute the labels of test set to validate that the DNN model is learning meaningful latent structure in terms of the relationship between inputs features and output labels.

B. Impact of Number of APs

Here we investigate how the performance of *E-app* varies with the number of available APs in the network. An accurate AP positioning prediction indicates that the predicted $\hat{Y} \in [0,1]^{(r_l \cdot r_w)/g_l}$ fully matches the ground truths. To relax the matching results, we define a error bar (EB) to accept the case when $\max_{i \in N}\{|P_i - G_i|\} \leq EB * g_l$, where g_l is the grid length and set as 0.2m, P_i and G_i are the predicted position and ground-truth position of AP_i, respectively. Fig. 8 shows the prediction accuracy vs. the number of deployed APs in a WLAN scenario, where the prediction model of *E-app* is trained with 20,000 data samples under environment dynamics. It is observed that the prediction accuracy of around 90%–98% is achieved for the cases with 1–5 APs and an EB of 2, which means that over 90% of predicted AP positions have the distance error of less than 0.4m to the ground truths. In particular, we note that *E-app* works fairly well when there are 1-3 APs deployed in the network, which is a common AP density in most small to medium-sized mmWave WLAN scenarios, and 91%–97% of predicted values can match the ground truths with EB of 1 (i.e., distance error of 0.2m).

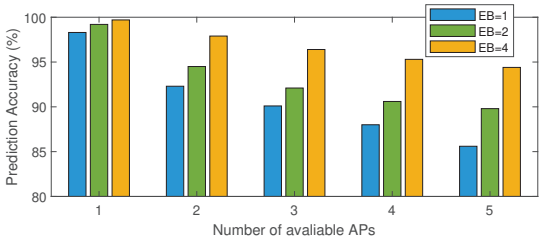


Fig. 8. Prediction accuracy vs. number of deployed APs.

C. Impact of Training Data Samples

Although *E-app* enables offline training process, the larger amount of data samples will consume more computational resources and training time that may affect the real-time online inference. Here we evaluate the prediction accuracy while increasing the number of data samples. Clearly, we can observe that there is a tendency that the prediction accuracy increases as the number of data samples increases, and the prediction accuracy for dataset from as low as 5000 data samples is also reasonably accurate. Specifically, the prediction accuracy increases from 84% to 94% (with EB of 2) as the number of data samples increases from 5,000 to 35,000, respectively. However, the achievable performance becomes saturated when the amount of data is beyond 20,000, since every additional 5,000 data samples only bring less than 1% performance improvement. This result implies that generating an appropriate amount of training data is critical to balance the prediction performance and the resource cost.

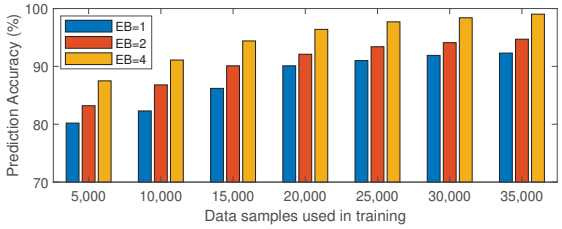


Fig. 9. Prediction accuracy vs. data samples used in training.

D. Impact of Obstacle Map

Next, we investigate the prediction performance of the proposed approaches over three different obstacle maps in 9m*6m*3m, 12m*8m*3m, and 16m*10m*3m rooms, respectively. We fix the training data samples at 20,000 and set EB as 2. Fig. 10 shows the prediction accuracy vs. 3 scenarios with different initial obstacle map and multi-AP placements. It is observed that the mean accuracy is 90% – 98% with a standard deviation of 0.52% – 2.44% for cases with different number APs. The low variance demonstrates that the proposed *E-app* is generalizable to different scenario instances, exhibiting the robustness of the proposed approach.

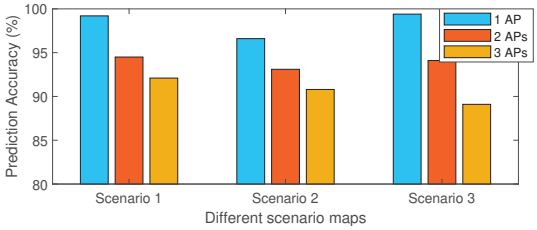


Fig. 10. Prediction accuracy vs. different scenarios.

E. Network Coverage under Environment Dynamics

In this part, we investigate the network coverage improvement brought by our proposed *E-app*. We define the network coverage rate (CR) as the fraction of LoS areas over the all critical viewpoints, i.e.,

$$CR = \frac{\sum\limits_{i \in VP} P_{w_i} \cdot I_{LoS}(i)}{\sum\limits_{i \in VP} P_{w_i}} \tag{13}$$

The results shown in Fig. 11 are generated by taking the average over different scenario cases. In each scenario, the original obstacles are randomly distributed within a WLAN scenario. Then, the environment dynamics are randomly added in every 3 minutes, including the operations of removing existing obstacles, adding new obstacles, or changing their locations, and P_w will also be changed accordingly. Here we add the AP planning scheme from [3] as the baseline scheme, which has shown to be superior to many state-of-the-art AP placement approaches such as [11]–[14]. In this situation, we observe that the network coverage is obviously improved along the timeline when adopting our *E-app* scheme, because the AP placements are adaptively updated to maintain

the optimal LoS performance when necessary. On the other hand, it is found that *E-app* can always detect the environment dynamics and make countermeasures adaptively (i.e., every 3 minutes), which implies the effectiveness of our sparse sampling approach to identify the critical OPs.

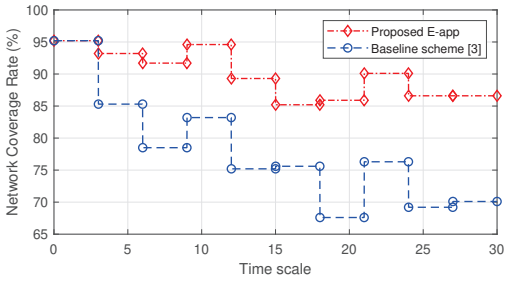


Fig. 11. Network coverage rate under environment dynamics.

F. Throughput Performance under Environment Dynamics

Similar to Sec. VII-E, here we consider 10 users moving around in the scenario, whose next destination point is determined by the popularity weights of pre-defined hotspot locations, and the mobile user moves on a straight-line path from the current location to the next one. Both the pause time at each destination point and the speed of movement follow the log-normal distribution, where the speed $ln(v) \sim \mathcal{N}$(-0.05, 0.69) (the mean is 1.21m/s) and pause time $ln(t_p) \sim \mathcal{N}$(3.15, 0.70) (the mean is 30.0s). The evaluations are done at the mmWave frequency of 60 GHz with a 2.16 GHz bandwidth, and the transmission power at AP side is 10 dBm. we adopt the channel model for indoor mmWave communication in [25], which is based on the single carrier PHY mode, supporting 12 modulation and coding schemes (MCS) [1]. Each data point, reported in Fig. 12, is the result of aggregate throughput over all users in the network. We observe that our *E-app* can significantly improve the user throughput performance by tuning the AP deployments to resist the environment dynamics, where 25.14% improvements are achieved as compare to the baseline scheme.

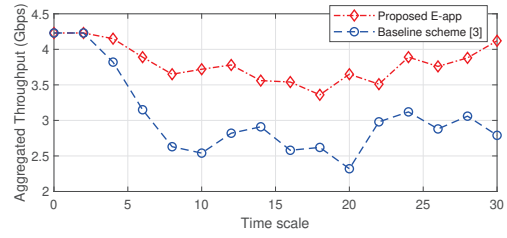


Fig. 12. Aggregate user throughput with *E-app* scheme.

VIII. Conclusion

In this paper, we studied adaptive AP planning in mmWave WLANs. By capturing the details of the environment and the sensitive locations to dynamics such as the change of obstacle layout and user-location popularity, we propose *E-app* to predict optimal AP placements when necessary. In particular, a learning-based model was trained to address the AP positioning problem under environment dynamics by using our synthetically generated dataset. Extensive evaluations were performed to show that our approach can achieve high prediction accuracy and provide significant improvement on network performance.

References

[1] IEEE Standard 802.11ad-2012. https://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=6392842

[2] IEEE Standard 802.11ay: *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications: Enhanced Throughput for Operation in License—Exempt Bands Above 45 GHz*, Dec. 2020.

[3] Y. Liu et al., "Maximizing line-of-sight coverage for mmWave wireless LANs with multiple access points", *IEEE/ACM Trans. on Netw.*, 2021.

[4] S. Chatterjee, M. J. Abdel-Rahman, and A. B. MacKenzie, "Robust access point deployment and adaptive user assignment for indoor millimeter wave networks", *IEEE Int'l Conference on Commu.*, 2020.

[5] Y. Liu, Y. Jian, R. Sivakumar, and D. Blough, "Optimal Access Point Placement for Multi-AP mmWave WLANs", *Proc. of ACM Modeling, Analysis and Simulation of Wireless and Mobile Systems*, 2019.

[6] T. Wei, A. Zhou, and X. Zhang, "Facilitating robust 60 GHz network deployment by sensing ambient reflectors", *Proc. USENIX Symposium on Networked Systems Design and Implementation*, 2017.

[7] A. Zhou, S. Xu, S. Wang, J. Huang, et al., "Robotic millimeter-wave wireless networks", *IEEE/ACM Transactions on Networking*, 2020.

[8] Y. Jian, M. Agarwal, S. K. Venkateswaran, et al., "Wimove: Toward infrastructure mobility in mmWave WiFi", *Proc. of ACM Symposium on Mobility Management and Wireless Access*, 2020

[9] M. Gowda, D. Ashutosh, and R. Choudhury, "The case for robotic wireless networks", *Proc. of Int'l Conference on World Wide Web*. 2016.

[10] Y. Jian, S. Lall, and R. Sivakumar, "Toward a self-positioning access point for wifi networks", *ACM MobiWac*, 2018.

[11] W. Li, L. Tian, J. Zhang, and Y. Cheng, "Analysis of base station deployment impact on LOS probability model for 5G indoor scenario", *Proc. of IEEE/CIC Int'l Conference on Commun. in China*, 2017.

[12] X. Qin, X. Yuan, Z. Zhang, F. Tian, Y. T. Hou, and W. Lou, "On AP assignment and transmission scheduling for multi-AP 60 GHz WLAN", *Proc. of IEEE Mobile Ad Hoc and Sensor Systems*, 2017.

[13] S. Sur, I. Pefkianakis, X. Zhang, and K.-H. Kim, "Towards scalable and ubiquitous millimeter-wave wireless networks," *Proc. of ACM International Conference on Mobile Computing and Networking*, 2018.

[14] R. Singh and D. Sicker, "SHINE (Strategies for High-frequency INdoor Environments) with Efficient THz-AP placement", *Proc. of IEEE Vehicular Technology Conference*, 2020.

[15] Z. Yang, C. Wu, and Y. Liu, "Locating in fingerprint space: Wireless indoor localization with little human intervention", *Proc. of ACM MobiCom*, 2012.

[16] B. Ferris, D. Fox, and N. Lawrence, "WiFi-SLAM using Gaussian process latent variable models," *Proc. of IJCAI*, 2007.

[17] Y. Zhu, Y. Zhu, B. Y. Zhao, and H. Zheng, "Reusing 60GHz radios for mobile radar imaging", *Proc. of ACM MobiCom*, 2015.

[18] F. Guidi, A. Guerra, and D. Dardari, "Millimeter-wave massive arrays for indoor SLAM", *IEEE Int'l Conference on Commun. Workshops*, 2014.

[19] P. Zhao, et al, "mid: Tracking and identifying people with millimeter wave radar", *Proc. of IEEE ICSCSS*, 2019.

[20] M. Lecci, P. Testolina, M. Polese, M. Giordani, and M. Zorzi, "Accuracy versus complexity for mmWave ray-tracing: A full stack perspective, *IEEE Transactions on Wireless Communications*, 2021.

[21] A. Zhou, et al, "Robot navigation in radio beam space: Leveraging robotic intelligence for seamless mmwave network coverage", *ACM Mobile Ad Hoc Networking and Computing*, 2019.

[22] M. Alrabeiah, A. Hredzak, and A. Alkhateeb, "Millimeter wave base stations with cameras: Vision-aided beam and blockage prediction", *Proc. of IEEE Vehicular Technology Conference*, 2020.

[23] S. Sur, I. Pefkianakis, X. Zhang, and K.-H. Kim, "Wifi-assisted 60 Ghz wireless networks", *ACM Mobile Computing and Networking.*, 2017.

[24] J. Choi, V. Va, N. Gonzalez-Prelcic, R. Daniels, C. Bhat, and R. Heath, "Millimeter-wave vehicular communication to support massive automotive sensing", *IEEE Communications Magazine*.

[25] Y. Liu and D. Blough, "Blockage robustness in access point association for mmWave wireless LANs with mobility", *IEEE LCN*, 2020.

[26] C. Wu, Z. Yang, Z. Zhou, et al., "PhaseU: Real-time LOS identification with WiFi", *IEEE Conference on Computer Communications*, 2015.

[27] R. Cowan, "Objects arranged randomly in space: An accessible theory", *Advances in Applied Probability*, vol. 21, no. 3, pp. 543–569, 1989.

[28] T. Bai, R. Vaze, and R. W. Heath, "Using random shape theory to model blockage in random cellular networks", *Proc. SPCOM*, 2012.

[29] M. Youssef and A. Agrawala, "The Horus WLAN location determination system", *Int'l conf. on Mobile systems, applications, & services*, 2005.