

Received March 5, 2020, accepted April 14, 2020, date of publication April 27, 2020, date of current version May 15, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2990631

Propagation-Model-Free Base Station Deployment for Mobile Networks: Integrating Machine Learning and Heuristic Methods

LINGCHENG DAI¹, (Graduate Student Member, IEEE),
AND HONGTAO ZHANG¹, (Senior Member, IEEE)

School of Information and Communication Engineering, Beijing University of Posts and Telecommunications, Beijing 100876, China
State Key Laboratory of Integrated Services Networks, Xidian University, Xian 710071, China

Corresponding author: Hongtao Zhang (htzhang@bupt.edu.cn)

This work was supported in part by the National Science Foundation of China under Grant 61971064, in part by the Beijing Natural Science Foundation under Grant 4202048 and Grant L182035, and in part by the Fundamental Research Funds for the Central Universities under Grant 2019PTB-008.

ABSTRACT As densification is the promising trend of future mobile networks, deployment of base stations (BSs) becomes increasingly difficult due to the laborious procedures in network planning; besides, unreasonable layout may lead to poor coverage performance. Hence, this paper firstly trains a propagation-model-free received signal strength (RSS) predictor based on machine learning (ML) models, and then optimizes coverage performance of BS deployment via multi-objective heuristic methods. Specifically, many practical features that affect signal propagation like geographical types and operating parameters of BS, are fed into ML models to predict RSS in a rasterized area; then based on the trained model, a well-designed multi-objective genetic algorithm (GA) is proposed to minimize the number of deployed BSs with coverage constraint. For the practical considerations of fast convergence and output-consistence, greedy algorithm with fixed initial solution and searching direction is also carried out. Moreover, the typical scenarios of incremental deployment (the mobile operator needs to deploy more BSs on the basis of the existing deployment) and BS outage compensation (one BS fails and other BSs need to adjust their configurations to fill the coverage gap), are also investigated for practical needs. Simulations show that multi-layer perceptron outperforms other ML algorithms in terms of RSS prediction with mean absolute error (MAE) yielded to 3.78 dB; and numerical results verify the convergence and availability of the proposed algorithms, which shows 18.5% gain than the real-world deployment in terms of coverage rate.

INDEX TERMS Propagation-model-free, base station deployment, machine learning, genetic algorithm, greedy algorithm.

I. INTRODUCTION

The exponential growth of mobile data traffic brought by smart devices and mobile Internet is approaching the capacity of current network infrastructures [1]. To address this challenge, ultra-dense network comes out as one of the leading concepts in the race to the future networks, where the basic idea is to provide a coverage environment so that users can connect to the access nodes as close as possible, i.e., base stations (BSs) are ultra-densely deployed in hotspots [2]. However, unreasonable layout of densely deployed BSs may

lead to poor coverage performance and extra capital expenditure (CAPEX) and operational expenditure (OPEX) [3]. Therefore, how to extend the wireless coverage by optimally deploying BSs while minimizing the number of BSs becomes an intractable problem in the field of BS deployment.

In general, the procedures of BS deployment can be divided into 3 steps: 1) determine the number and the address of new BSs according to users' needs; 2) further decide the exact locations of BSs by alternating optimization between field measurements and coverage evaluation via propagation models; 3) start deployment. As we can see, the procedures of BS deployment is time-consuming and costly. Besides, there exists many challenges during practical deployment of BSs:

The associate editor coordinating the review of this manuscript and approving it for publication was Xiaofan He¹.

- Many existing empirical radio propagation models, such as Okumura-Hata [4] and COST [5], may become inaccurate or not longer applicable due to the change of geographical environment and spectrum [6]; besides, many factors that influence the signal propagation like geography types are not considered.
- Although there are some network planning tools that focus on radio frequency coverage planning (e.g., CelPlan [7]), they cannot avoid the intricate procedures in building an approximator of propagation models for received signal strength (RSS) prediction, and require signal measurements in the field.
- Current network planning is an relatively independent process and lots of experience as well as useful information is underutilized, which implies that there is significant potential gain remaining to be mined.

In addition, mobile operators have to handle many situations more than just deploy BSs in an unplanned area, such as: 1) incremental deployment, i.e., there have already existed some deployed BSs and the operator needs to deploy more to increase the coverage; 2) BS outage compensation, i.e., a BS fails to provide coverage to its users and neighbor BSs adjust their configurations (e.g., antenna tilt or transmit power) to fill the coverage gap caused by this faulty BS. Adapting different solutions to solve these challenges is complicated and costly, which drives self-organizing network (SON) [8] to reduce human intervention. However, there still exist many different SON functionalities focused on solving various situations nowadays, which is not cost-saving. Therefore, it is necessity to design a systematic technique to extract relevant information from lots of valuable information and diminish operational costs.

In this context, machine learning (ML) is utilized as a powerful tool to extract relevant information from the rich cellular data. ML has achieved great improvements and extensive applications nowadays, however, there are only a few studies to analyze effective RSS prediction model and optimization of BS deployment for the purpose of achieving better coverage performance and lower overhead.

A. RELATED WORKS

Path loss models are extensively used in signal prediction and coverage evaluation, which is crucial for BS deployment [9]. The design of mobile communication networks requires a good knowledge of wireless channel. Many researches have focused on designing path loss prediction models based on drive test measurements [10], [11]. The authors of [12] developed a neural network for path loss predictions with normalized terrain profile data. And regression algorithms were used in [13] by extracting relevant information from the huge amount of radio measurements for quality of service (QoS) prediction. However, many important factors that affect the signal propagation, such as geography types, BS height, downtilt of antennas and azimuths, are not taken into the consideration of signal prediction. In our previous work

[14], a propagation-model-free coverage evaluation model based on ML was proposed to predict the received signal strength (RSS) at each grid, where the majority of important factors were considered in order to improve the prediction accuracy; however, the problem was modeled as a classification problem, which is not in line with the nature of RSS prediction. Moreover, how to further optimize the BS deployment is not discussed in those works above.

Theoretical algorithms were proposed in [15]–[17] to address the nonconvex and combinatorial optimization problem of BS deployment. Nevertheless, these works assumed simple propagation model and did not consider practical scenarios of BS deployment, which cannot be applied in practical systems. In [18], different works that aim to solve the problem of QoS prediction and verification was offered, but it directly focused on QoS offered to end-users and the resources that the operators need to offer, rather than optimized the coverage performance of networks.

In [19], a pico BS deployment problem was formulated as an additional part to meet the increasing data exchange requirements, which assures the performance of coverage and quality of services; besides, [20] proposed many BS deployment algorithms including region-based, grid-based and greedy algorithms to determine the most suitable positions of micro BSs. However, these works only consider the impact of location, where other parameters that affect the performance indicators are not taken into consideration. Moreover, those algorithms optimized only one variable in each iteration and was performed in an exhaustive manner, which is inefficient with poor performance.

B. CONTRIBUTIONS

Therefore, different from aforementioned works, our proposed BS deployment method firstly extracts main features that determine the strength of signal propagation from tremendous RSS data. Then ML techniques are leveraged to train a regressor for RSS prediction. Moreover, in order to address the non-convex optimization problem of BS deployment, a multi-objective genetic algorithm (GA) and greedy algorithm are conducted. The locations and operating parameters of deployed BSs are optimized to minimize the number of BSs with coverage guarantee. In this paper, homogeneous BSs are considered to focus on the design of BS deployment algorithms, where other types of BSs can be deployed layer-by-layer with the same technique. Moreover, since coverage performance is priority during initial network planning, QoS demand is not taken into consideration, which can be optimized in future research.

The main contributions of this paper can be summarized as follows:

- Different from deriving empirical propagation models from field measurements, the knowledge of channel features (including topographic types and operation parameters of BSs) from different locations are extracted via ML-based propagation-model-free method, which

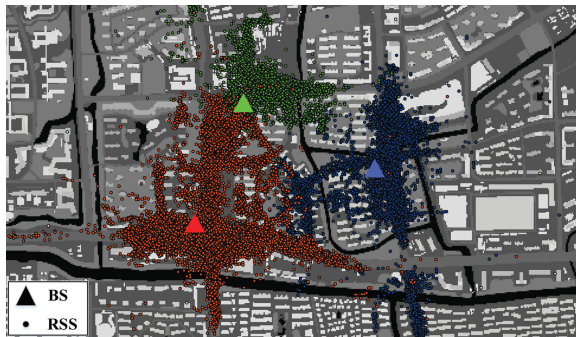


FIGURE 1. Screenshot of dataset consists of several outdoor BSs as well as their corresponding RSS in a real-world area. Triangle represents the BS, and colored dots represent the RSS belong to different BSs. The darker is the dots, the higher strength of received signal they represent.

makes it easy to generalize to other locations without too much tuning and saves a large amount of CAPEX and OPEX.

- Multi-objective genetic algorithm is proposed to solve the minimization problem of the number of deployed BSs by optimizing the locations and operating parameters. The constraints of coverage, BS deployment terrain and operating parameters range are taken into account for practical systems. Moreover, algorithm with fixed initial solution and optimization direction is also carried out under the practical considerations of fast convergence and the output-consistent.
- In addition to deploying BSs in an unplanned area, the proposed BS deployment algorithms can still be used as SON functionalities in the typical scenarios like incremental deployment and BS outage compensation. As a result, it can be used as a substitute for the network management tool, which reduces lots of operational costs for mobile operators.

The rest of the paper is organized as follows. Section II introduces the data needed to be collected for training and formulates the optimization problem. Then the network planning tool including offline training, online evaluation and practical applications is described in Section III. Section IV presents the details of the simulation, as well as the meaningful numerical results. Finally, Section V summarizes with the conclusions.

II. PRELIMINARY

A. DATA COLLECTION FOR MODELLING RSS ESTIMATION

In order to accurately estimate RSS received at the user side, main factors that affects signal propagation should be taken into consideration as many as possible. Among all the factors, the most dominant one is the distance between BS and user due to the principle of electromagnetic propagation. The power strength will decrease as the distance from source increases (known as the Inverse-square Law). Besides, geography is also an important element in affecting signal propagation especially in the city. Signal propagation will be blocked by mountains, huge buildings and other ground features, which obstructs line of sight transmission and lead to reflection and diffraction. In addition, another important

factor is the operating parameters of BS, such as transmit power, height, azimuth and mechanical (electrical) downtilt.

Fig. 1 shows a part of RSS distribution map as well as several different BSs in a real-world area. Note that the distribution of RSS is discrete and they may overlap with each other, which increases the difficulty in data analysis and channel modelling.

We build a data set collected from the real-world networks of multiple locations. The data contains samples (rows), and features (columns), which can be divided into 2 sets—the training set and the test set. The training set is used to train the model while the test set is used to evaluate how much the predictions are correct. Supervised machine learning is utilized to develop a predictive model by deducing a specific function $f(\mathbf{x})$ from the output \hat{y} . Assume there are n selected features, then a training sample can be represented by a n -dimensional input vector $\mathbf{x} = (x^{(1)}, \dots, x^{(n)}) \in \mathbf{R}^n$. Assume there are p training samples $((\mathbf{x}_1, y_1), \dots, (\mathbf{x}_p, y_p))$ in a training set. Each training sample attaches to a corresponding output, i.e., RSS value.

According to the feature selection criteria discussed earlier, a total of 25 features that affects signal propagation is collected to learn the relationship between input and output. The features required are listed in Table 1 as follows:

TABLE 1. Required features from real-world datasets.

Symbols	Description
d	Distance between grid and BS
Ψ	Geography statics collection
$\hat{\lambda}_{azi}$	The collection of three azimuths of BS
λ_{MD}	Mechanical downtilt of BS
λ_{ED}	Electrical downtilt of BS
h	Height of BS
P	Transmit power
rss	Received signal strength

The first feature is the distance between user and BS, denoted by d in meters. $\Psi = [\varphi_1, \dots, \varphi_K]$ is a vector that calculates the main geography statistics between users and BSs, where φ_i represents the proportion of i -th geography types, such as buildings, mountains, forest and so on. After the comparisons of feature importance and manual screening, we select a subset of geography features with $K = 17$ from the 3D map dataset. $\hat{\lambda}_{azi} = [\lambda_{azi}^{(1)}, \lambda_{azi}^{(2)}, \lambda_{azi}^{(3)}]$ represents collection of the three azimuths of BS in a cellular network. λ_{MD} and λ_{ED} represent mechanical downtilt and electrical downtilt, respectively. And h is the relative height of BS, P is the transmit power of BS measured in dBm.

B. PROBLEM FORMULATION

The task in this paper is to meet the coverage requirement by deployed as less BSs as possible. In order to calculate the coverage rate, a given region is rasterized uniformly into L grids with size of $20\text{m} \times 20\text{m}$. For example, a $500\text{m} \times 1000\text{m}$ area can be divided into 25×50 grids. Then for each grid i , we have a RSS predicted by supervised learning model, denoted by rss_i . Therefore, given the collected data for one BS

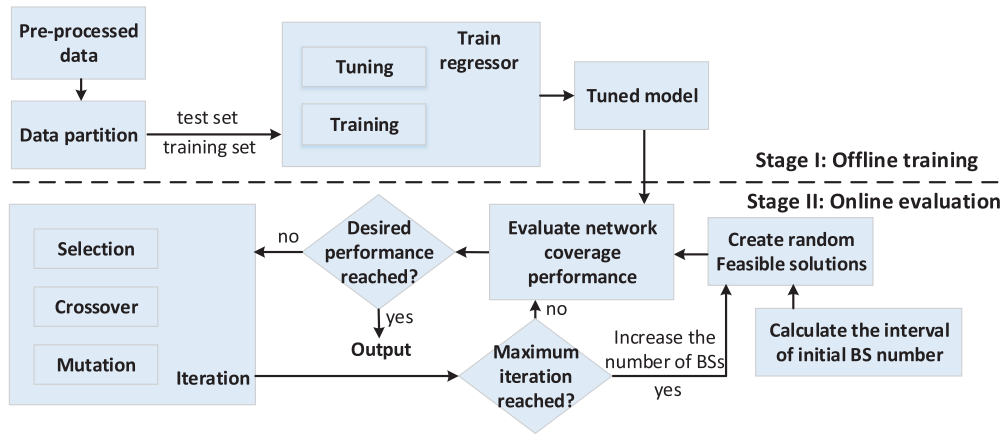


FIGURE 2. Architecture of network planning tool, including two stages: offline training and online evaluation. Offline training is to train a RSS predictor regressor for the coverage evaluation in stage II; online evaluation leverages heuristic approaches to minimize the deployed number of BSs.

with size of $[L \times n]$, the model produces an output with size of $[L \times 1]$. Note that each grid may have multiple RSS values corresponding to different BSs, the largest RSS is selected as the indicator of coverage.

In order to formulate the optimization problem in a mathematical form, we introduce a variable, \mathbb{C} , which indicates whether the predicted RSS of a grid meets the coverage requirements:

$$\mathbb{C}_i = \begin{cases} 1, & \text{if } rssi_i \geq \phi \\ 0, & \text{if } rssi_i < \phi \end{cases} \quad (1)$$

where ϕ is a predefined RSS threshold. Let \mathcal{C} denote the coverage rate, then the definition is given by:

Definition (Coverage rate): The coverage rate \mathcal{C} of a specific area with several BSs is defined as the ratio of the number of grids greater than a certain threshold C_{th} to the total number of grids, i.e., $\mathcal{C} = \sum_{i=1}^L \mathbb{C}_i / L$.

The goal is to minimize the number of deployed BSs while ensuring the network coverage. Denote the set of deployed BSs as $\mathcal{A} = \{1, 2, \dots, A\}$, where A is the number of BSs. Therefore, the multi-objective optimization problem can be formulated as follow:

$$\begin{aligned} \min_{\{\theta\}_{a \in \mathcal{A}}} \quad & A \\ \text{s.t.} \quad & C1 : \mathcal{C} \geq C_{th} \\ & C2 : (G_x, G_y) \in \mathbb{R}_d \\ & C3 : \lambda_{azi}^{LOW} \leq \lambda_{azi}^{(i)} \leq \lambda_{azi}^{UPPER}, i = 1, 2, 3 \\ & C4 : \lambda_{MD}^{LOW} \leq \lambda_{MD} \leq \lambda_{MD}^{UPPER} \\ & C5 : \lambda_{ED}^{LOW} \leq \lambda_{ED} \leq \lambda_{ED}^{UPPER} \\ & C6 : h^{LOW} \leq h \leq h^{UPPER} \\ & C7 : P^{LOW} \leq P \leq P^{UPPER} \end{aligned} \quad (2)$$

where $\theta = (n, G_x, G_y, \hat{\lambda}_{azi}, \lambda_{MD}, \lambda_{ED}, h, P)$ denotes the seven optimized parameters of the BS, i.e., (G_x, G_y) represents longitude and latitude of the BS to be deployed. (C1) is the coverage constraint over the whole area. \mathbb{R}_d indicates the planned area that is suitable for BS deployment such

as the tall buildings, mountains and flat land, so that (C2) indicates that the solutions of longitude and latitude must be in the region. Let $(\bullet)^{LOW}$ and $(\bullet)^{UPPER}$ represent the lower bound and the upper bound of corresponding parameters, respectively, where the lower bound and upper bound are statistics from actual data sets. Therefore, (C3–C7) represent the constraints of azimuths, mechanical downtilt, electrical downtilt, BS height and BS power, respectively. Note that the optimization problem contains many real-world constraints that cannot be modeled, it cannot be solved by numerical optimization. Therefore, we seek to solve this problem with heuristic approaches.

III. NETWORK PLANNING TOOL

In this section, we aim to create a novel BS deployment tool, which solves the optimization problem (2) and achieves the same coverage performance as in existing LTE network. However, accurate coverage evaluation is difficult due to the complex propagation environment; besides, the space of feasible solution is incredibly large, which makes the global optimal solution difficult to obtain. Therefore, we divided the network planning tool into two stages. Stage I is responsible to train a regressor by means of supervised learning approach [21], which can be done offline. And Stage II leverages the output model of Stage I to estimate the coverage rate for a feasible solution. Then based on this evaluation function, the parameters can be optimized to find the best solution (i.e., location and operating parameters of BSs) via heuristic algorithms such as genetic algorithm and greedy algorithm. The architecture of network planning tool is depicted in Fig. 2.

A. OFFLINE TRAINING

Supervised learning techniques can be generally classified into classification and regression, which depends on the predicted values are discrete or continuous. As mentioned above, the RSS prediction is a regression task. In this section, ML-based model is responsible to learn the function $f(\mathbf{x})$ that represents the relationship between the features \mathbf{x} and the RSS value y . Therefore, the exactitude of coverage evaluation highly depends on the accuracy of RSS prediction.

Algorithm 1: Train Regressor

Input: The processed dataset $(X_{\text{train}}, Y_{\text{train}})$ and $(X_{\text{test}}, Y_{\text{test}})$, Maximum number of iterations T

Output: Model

Given the training set $X_{\text{train}} = \{x_1, \dots, x_p\}$,
 $Y_{\text{train}} = \{y_1, \dots, y_p\}$;
 \\Choose the best values by using grid search;
for $t = 1 : T$ **do**
 | \\Call regressors
 | $\text{model}(t) := \text{regressor}(X_{\text{train}}, Y_{\text{train}}, \text{parameters});$
 | \\ Predict the RSS value:
 | $\hat{Y}_{\text{test}} := \text{model}(t). \text{predict}(X_{\text{test}});$
 | \\Evaluate performances by comparing with the
 | actual value (Y_{test})
 | $\text{Score} := \text{model}(t). \text{score}(Y_{\text{test}}, \hat{Y}_{\text{test}});$
end
return $\text{model} := \text{best}(\text{model})$

In order to improve the prediction performance, we exhaustively compare several ML algorithms such as k -nearest neighbor, random forest, SVM with different kernels and multi-layer perceptron (MLP). The raw data collected in Section II is firstly processed by data cleaning, normalization, dimensionality reduction. Then the processed data is split into training set and test set. Based on the training set, each ML algorithm traverses all the parameter settings and find the optimal parameters that obtains the best prediction performance. Among all the ML models, we choose the model that fits the data best to evaluate coverage in Stage II. The detailed training process is described in Algorithm 1.

1) DATA PRE-PROCESSING

a: DATA CLEANING

Data mining techniques highly rely on a clean and integral dataset. However, faulty sampling process and limitations of the data acquisition process will lead to missing values. Many approaches are available to handle those samples, such as expectation-maximization (EM), multiple imputation [22]. In this work, the missing values are discarded for simplicity.

b: NORMALIZATION

The range of values of raw data varies widely, which may make some ML algorithms not work properly and leads to slow convergence. Therefore, data normalization, which is also known as feature scaling, is used to normalize the range of features of data. In this work, min-max normalization is adopted in order to rescale the feature values to the range of $[0, 1]$:

$$x' = \frac{x - \min(x)}{\max(x) - \min(x)} \quad (3)$$

c: DIMENSIONALITY REDUCTION

Principle component analysis (PCA) [23] is adopted to reduce the feature dimensions, which can improve the training effi-

ciency and performance. The main idea is to decompose a multivariate dataset into a set of successive orthogonal components that explain a maximum amount of the variance. It is essential in practical implementation especially when the input data is with huge amount of features. The total number of feature dimensions in our dataset 25, therefore, PCA is adopted.

2) DATA PARTITION

The pre-processed data is split randomly into training set $(X_{\text{train}}, Y_{\text{train}})$ and test set $(X_{\text{test}}, Y_{\text{test}})$. The training set contains p samples and the test set contains $m = L - p$ samples, where approximately $p : m = 7 : 3$. The test set is utilized to select the best model, the model will not be tuned any further once the model has been well trained.

3) HYPERPARAMETER OPTIMIZATION

Due to the large number of parameters embedded in regressor algorithms, finding the setting of regressor that fits the data best is difficult. Therefore, we only consider the main parameters in each algorithm, such as C and ϵ in SVM, k in nearest neighbors [21]. Then exhausted grid search algorithm is applied to perform hyperparameter optimization.

4) EVALUATION OF PREDICTION PERFORMANCE

After the model is tuned, the test set can be used to evaluate the performance of the tuned model. For each predicted value, we evaluate the performance against the actual value in terms of the mean absolute error (MAE) as follows:

$$MAE = \frac{1}{m} \sum_{i=0}^{m-1} |\hat{y}_i - y_i| \quad (4)$$

where \hat{y}_i and y_i indicate the predicted value and the testing actual value of the i -th data sample, respectively. m is the size of the test set.

B. ONLINE EVALUATION BY GENETIC ALGORITHM

In this section, we propose a multi-objective genetic algorithm to optimize the locations and operating parameters of deployed BSs on the basis of coverage evaluation function, which is shown in the Stage II of Fig 2. The proposed GA performs parallel search from a population and avoids local optimum by following probabilistic rather than deterministic search rules. In particular, we first calculate the interval of the number of BSs to be deployed and try the GA from the minimum value of the interval. If coverage meets the requirements, then the optimal solution is output with minimum number of BSs; otherwise, add a BS and perform GA again. In each iteration of GA, a population of feasible solutions are generated randomly and the relevant data is collected to evaluate their coverage respectively. Among all the individuals in the population, we select several best individuals according to a certain selection method and exchange their chromosomes, namely the crossover. To escape the local optimum, gene mutation is adopted. More details are depicted in Algorithm 2.

Algorithm 2: Multi-Objective Genetic Algorithm

Input: Trained regressor, *Regressor*,
 Initial population \mathbf{S} , size of population N ,
 Number of generations g ,
 Rate of elitism e , rate of mutation δ
 Predefined coverage rate C_{th} , RSS threshold ϕ

Output: Optimal solution $\hat{\theta}^*$

Given planning area, calculate \bar{a}

for $a = \lfloor \bar{a}(1 - 30\%) \rfloor, \dots, \lceil \bar{a}(1 + 30\%) \rceil$ **do**

\\Initialization

Create random feasible population

$\mathbf{S} = \{\hat{\theta}^1, \hat{\theta}^2, \dots, \hat{\theta}^N\}$

for $i = 1 : g$ **do**

forall $\hat{\theta} \in \mathbf{S}$ **do**

Collect necessary information X for prediction

\\Predict and count the RSS that larger than ϕ

$\text{Fitness}(\hat{\theta}) := \text{Regressor}(X)$

if $\text{Fitness} \geq C_{th}$ **then**

return the best solution $\hat{\theta}^*$

Break

end

end

\\Selection based on Tournament Selection

Select the best e individuals

\\Crossover based on Uniform Crossover

Number of crossover $n_c = e/2$

for $j = 1 : n_c$ **do**

Randomly select two individuals $\hat{\theta}^a$ and $\hat{\theta}^b$

Generate $\hat{\theta}^c$ by Uniform Crossover

\\Mutation

Mutate each parameter of $\hat{\theta}^j$ under the rate δ

end

end

end

return the best solution $\hat{\theta}^*$

1) CREATE RANDOM FEASIBLE SOLUTIONS

We create a set of random feasible solutions (also called chromosomes or individuals), $\mathbf{S} = \{\hat{\theta}^1, \hat{\theta}^2, \dots, \hat{\theta}^N\}$, where N is the population size. As the number of BSs deployed in a given area is uncertain and the search space can be very large, an integer value \bar{a} is firstly derived by counting the number of BSs in the existing cellular network, where the range of the initial number of planned BSs a is $\lfloor \bar{a}(1 - 30\%) \rfloor, \dots, \lceil \bar{a}(1 + 30\%) \rceil$, 30% is an empirical value. Therefore, the parameter vector of an individual n can be denoted by $\hat{\theta}^n = (\theta_1^n, \theta_2^n, \dots, \theta_a^n)$.

It is worth noting that many geography types are not suitable for BS deployment, such as inland water, wet land, forest and so on. Therefore, we make statistics on the types of geographies that are suitable for building stations and mark those terrains that are not. Then the feasible solutions are generated only on the suitable terrains, which

helps to find better solution and fasten the convergence procedure.

2) EVALUATE THE COVERAGE PERFORMANCE

The objective of this module is to design a function (also called fitness) to evaluate the coverage of each individual. Given a feasible solution $\hat{\theta}^n$, this function is responsible for predicting the values of all grids by the model produced during the offline training, and returning the coverage rate of $\hat{\theta}^n$. Specifically, in each iteration we firstly collect the required data of all grids for each feasible solution, $\hat{\theta}$. Then the collection is input to the trained regressor to obtain the RSS value in each grid of the area. Each grid will have several RSS values attached to different BSs, and the maximum RSS value is selected as the indicator of this grid for the calculation of coverage.

3) SELECTION

This module generates a new population of individuals by selecting the best fit individuals from the current population for reproduction, which is known as elitist selection. Elitism chooses the best e fittest candidates into the next generation. There are many selection operators like proportionate roulette wheel selection, tournament selection and linear or exponential ranking selection [24]. In this work, tournament selection is used because it is easy to implement, which selects the best two for reproduction from population.

4) CROSSOVER

This function generates new offspring by inheriting part of the genes from their parents selected by elitist selection. There are also many genetic operators in terms of crossover, like one-point or two-point crossover, uniform crossover and arithmetic crossover. In this work, Uniform Crossover is used. The idea behind this operator is to combine the genes into one chromosome from two parents with a mixing ration. Unlike one-point and two-point crossover, the uniform crossover means that genes at each chromosome of two matched individuals are swapped with the same crossover probability. If the mixing ratio is set to 0.5, the offspring owns approximately half of the genes from the first parent and the half from the other parent.

5) MUTATION

In order to avoid premature convergence on a local optimum, mutation is essential to maintain the diversity in the value of the parameters for next generation. Therefore, this module selects a uniform random value between the minimum and maximum value. The probability of mutation in a feasible solution $\hat{\theta}^n$ is set to δ . δ need to be chosen carefully because if δ is too high, the convergence is slow, otherwise it will converge to a local optimum. Finally the new individuals replace previous ones in the population.

C. ONLINE EVALUATION BY GREEDY ALGORITHM

Genetic algorithm can avoid local optimum due to its stochastic mutation, which also makes it difficult to converge;

besides, the global optimum is hard to attained in large planning region, which may lead to different outputs when the genetic algorithm is performed several times. From the perspective of industrial application, greedy algorithm with fixed the initial solution and search direction is suitable due to the advantages of fast convergence and output-consistence. In particular, we also set an interval of the number of deployed BSs as $\lfloor \bar{a}(1 - 30\%) \rfloor, \dots, \lceil \bar{a}(1 + 30\%) \rceil$. The greedy algorithm is performed with the number of deployed BSs ranging from the lower bound of the interval to the upper bound, until reaching the stop criteria that the coverage requirement is met. In each iteration, we create the fixed initial solution $\hat{\theta}^0$ according to the coordinates of the planning region. Then for each optimized parameter, a set of adjacent solutions is built by adding or subtracting their step size $\Delta = (\Delta_1, \Delta_2 \dots, \Delta_v)$, respectively. Among the adjacent set, we evaluate their coverage rate and greedily select the solution with maximum coverage as a new initial solution. If the coverage meets the requirements, break; otherwise, repeat the step in each iteration. The online evaluation by greedy algorithm is given in Algorithm 3.

D. USE CASES

1) INCREMENTAL DEPLOYMENT

It is a very typical application scenario that there have already existed some deployed BSs and the operator wants to deploy more for larger coverage. In this scenario, the proposed algorithms are also applicable. Take the parameters of existed deployed BSs as part of the chromosome, and generate the incremental BSs according to the procedures according to Algorithm 2 or Algorithm 3. This is for the case that there exist no other candidate BS depositories that have the tower but not at working. As for the case that there are some inactive BSs besides those active BSs, it is cost-saving to activate those BSs rather than build a new one. In this case, when initialize the feasible solution, the parameters of active BSs are fixed in the chromosome; for those inactive candidate BSs, fix their longitude and latitude and then update other parameters. Then the proposed model can be used to optimize the result. Therefore, the incremental deployment can be integrated as a global optimization problem as blank deployment.

2) BS OUTAGE COMPENSATION

During the network operation, the BS may be not able to provide services to its users within a certain period. Therefore, it is necessary to provide a tool that alleviates the outage via readjusting the parameters of nearby BSs to fill the coverage gap. Assume there is a BS corrupted suddenly, we fix the longitude and latitude of other normal BSs, then remove the faulty BS from the initial solution and maintain the coverage rate unchanged. Run the model until it satisfies the stop condition, the neighbor BSs will adjust their antenna tilt and transmit power to fill the coverage gap caused by the faulty BS. The simulations of these use cases are presented in the next section.

Algorithm 3: Greedy Algorithm-Based BS Deployment Optimization

Input: Trained regressor *Regressor*,
Number of optimized variables v ,
Initial solution $\hat{\theta}^0 = (\theta_1^0, \theta_2^0, \dots, \theta_v^0)$,
Step size $\Delta = (\Delta_1, \Delta_2 \dots, \Delta_v)$
Predefined coverage rate C_{th} , RSS threshold ϕ

Output: Optimal solution $\hat{\theta}^*$

Given planning area, calculate \bar{a}

for $a = \lfloor \bar{a}(1 - 30\%) \rfloor, \dots, \lceil \bar{a}(1 + 30\%) \rceil$ **do**

Create fixed initial solution $\hat{\theta}^0$

Assign the current solution to the optimal solution,
i.e., $\hat{\theta}^* = \hat{\theta}^0$

while $Fitness(\hat{\theta}^*) < C_{th}$ **do**

By adjusting the step size in each variable, attain many adjacent points Π

forall $\hat{\theta} \in \Pi$ **do**

Collect necessary information X for prediction

\\ Predict and count the RSS that larger than ϕ

$Fitness(\hat{\theta}) := Regressor(X)$

end

$\hat{\theta}^* = \arg \max_{\hat{\theta} \in \Pi} (Fitness(\hat{\theta}_i))$

if $Fitness(\hat{\theta}_i)$ remains unchanged over multiple iterations **then**

| break

end

end

end

return the best solution $\hat{\theta}^*$

TABLE 2. Parameters in algorithms.

Parameters	Value
Type of encoding	positive real value
Number of generations g	50
Size of population N	20
Elitism e	2
Rate of mutation δ	0.1
Step size Δ	0.01

IV. NUMERICAL RESULTS

In this section, numerical results are analyzed through the proposed network planning tool in Hangzhou, China. The dataset consists of over 760,000 samples from about 500 BSs, and each BS is attached with about 2000 RSS values in average. The threshold ϕ and C_{th} is -90 dBm, 70% if not specified. An area of 1 square kilometer will deploy an average of 4 base stations, i.e., $\bar{a} = 4$. The goal is to optimize the parameters (i.e., number of BSs, longitude, latitude, azimuths, mechanical downtilt, electrical downtilt, height and transmit power of each BS) that satisfy the coverage requirement (among all the grids, the RSS that larger than ϕ accounts for over C_{th}). More parameters used in the simulations of genetic algorithm are given in Table 2.

TABLE 3. Performance comparison of different RSS prediction schemes.

Algorithm	MAE	Training time(s)	Main parameters
Nearest neighbors	5.1233	6.630	$k = 4$
Linear SVM	6.0470	120.002	$C = 1$
RBF SVM	5.4810	180.221	$\gamma = 1, C = 1.5$
Decision tree	5.5217	6.093	max_depth=5
Random forest	5.7614	5.390	max_depth=5, n_estimators=10, max_features=1
Multi-layer perceptron	3.7883	60.371	$\alpha = 1, hidden_layer_size=100$
XGBoost	4.621	36.242	max_depth = 10, max_leaves = 1400
Feed-forward neural network	6.3819	24.512	Tangent activation, 48 neurons, 6 features [12]
Hata	19.862	—	[4]
COST 231	17.297	—	[5]

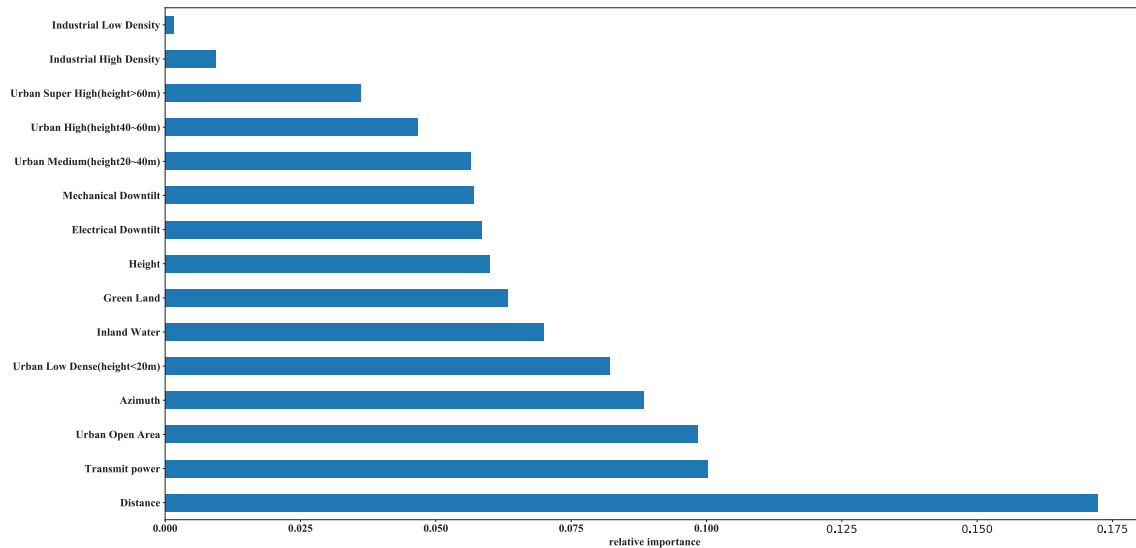


FIGURE 3. Feature importance measured via XGBoost.

Table 3 presents the prediction performance and the major optimized parameters for different RSS prediction models (i.e., k -nearest neighbor, SVM with linear and RBF kernel, decision tree, random forest, multi-layer perceptron, ANN [12], Hata [4], and COST 231 [5]). Among the parameters, k is the number of neighbors; C denotes the penalty parameter and η is the kernel coefficient; α and β represent the regularization term and learning rate in the multi-layer perceptron, respectively. As we can see, multi-layer perceptron outperforms other algorithms in terms of MAE with training time of 60.371 s in average. More complex models are more expensive to train, but the results are not necessarily good. The proposed regression schemes outperform ANN in [12] because more relevant features are used to train our model. All ML methods achieve much gain compared to empirical models. In this paper, multi-layer perceptron is leveraged to train the regressor and further used as a tool to evaluate the coverage performance in Stage II.

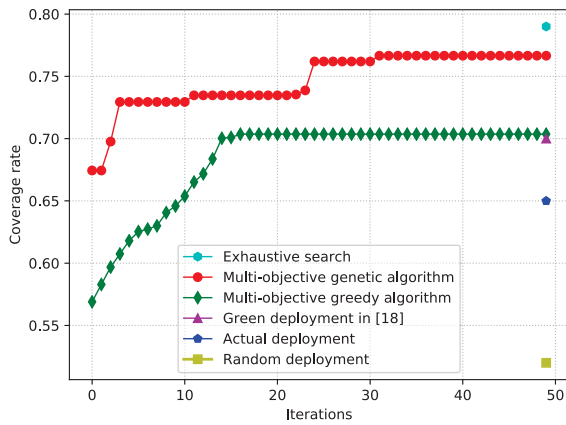
The impact on MAE of feature dimensions and data size is investigated on three main regression algorithms (kNN, SVM, MLP) in Table 4. The amount of features is 25, including distance, geography types and operating parameters of

TABLE 4. MAE comparison of different regression algorithms for different dimension and training set size.

MAE	kNN			SVM			MLP		
	2D	3D	25D	2D	3D	25D	2D	3D	25D
200,000 samples	8.10	7.37	5.98	7.91	7.55	6.05	8.69	7.69	7.14
400,000 samples	7.28	6.06	5.27	7.14	6.39	5.69	7.88	5.88	5.64
760,000 samples	6.42	5.87	5.12	6.78	5.73	5.48	6.94	4.21	3.79

BS, where the combination and reduction of these features have been tested to improve the prediction. PCA is conducted by re-scaling the 17 geography types and 4 operating parameters into 1 principle component, respectively. 2D refers to transforming distance as one dimension, other features as the other dimension. It is expected that with the increase of data size, the MAE of test set is improved. However, although best performance is achieved without PCA, the training cost is expensive and the importance of distance is overwhelmed. Therefore, in the following steps, 3D dataset is used to train the MLP.

Fig. 3 gives part of the feature importance ranking among all features measured through XGBoost. In the decision tree,



(a)

Iter. / Algo.	Num.	1 BS	2 BSs	3 BSs	4 BSs
Exhaustive search		—	—	✓	
Genetic Algorithm		[0, 10]	[11, 23]	[24, 30]	[31, 49]
Greedy Algorithm		[0, 7]	[8, 13]	[14, 49]	
Green deployment [18]		—	—	—	✓
Actual Deployment		—	—	—	✓
Random deployment		—	—	—	✓

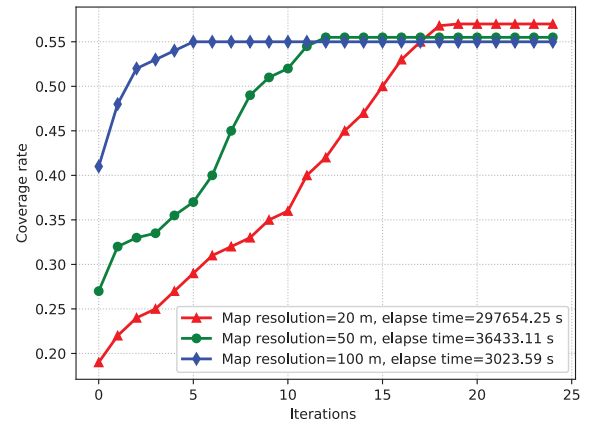
(b)

FIGURE 4. Evolution of different criteria with the increase of iterations for multiple BS deployment algorithms (a) the coverage rate; (b) the number of BSs.

the attribute value is calculated by improving the amount of performance variables at each attribute division point, which is responsible and recorded times. And the result of one attribute in all the promotion trees is summed up as expected and then averaged to obtain the degree of importance. Therefore, it can be seen that distance between the BS and grid plays the most important part in predicting RSS, accounting for about 0.17 in terms of relevant importance. Transmit power ranks second and other parameters of BS also play a big role, which is as expected. Buildings have big accounting because our dataset comes from cities in the land so that the features of sea, wet land and forest almost have no impact on predictions.

Fig. 4 shows the evolution of average coverage rate and the number of BSs with the increment of iterations for different BS deployment algorithms in an unplanned location with size of 1km×1km. For benchmarks,

- 1 Exhaustive search, where grid search is performed on all the parameters within the range to find the best parameters that meet the coverage requirement;
- 2 Green BS deployment [19], in order to make the algorithm suitable for this paper, we fix the other parameters except the locations and coverage assurance phase in Algorithm 3, [19] is perform, which BS is added one-by-one greedily;
- 3 Random deployment, where all parameters are generated randomly.

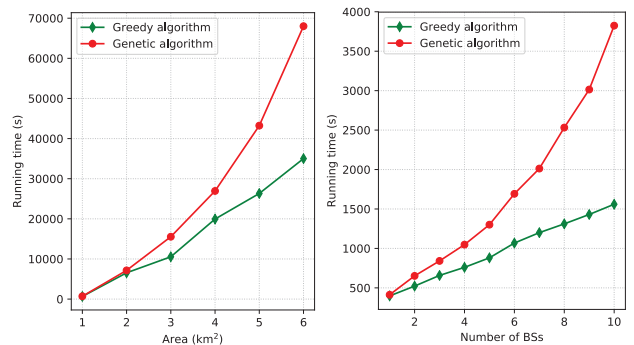


(a)

Iter. / Reso.	Num.	4 BSs	5 BSs	6 BSs
20 m		[0, 10]	[11, 17]	[18, 24]
50 m		[0, 6]	[7, 10]	[11, 24]
100 m		[0, 1]	[2, 4]	[5, 24]

(b)

FIGURE 5. Evolution of different criteria with the increase of iterations under different map resolution in a 3 × 3 km2 region (a) coverage rate; (b) the number of BSs.



(a) $\phi = -90$ dBm, $C_{th} = 80\%$ (b) $\phi = -90$ dBm, $C_{th} = 80\%$, area = 1 km²

FIGURE 6. Comparison of convergence time between the two algorithms under different planning area and number of BS deployments.

where all the schemes need to produce feasible solutions that satisfy the geography constraints. The convergence of two algorithms proposed in this paper is validated, and genetic algorithm outperforms the greedy algorithm in terms of coverage rate because it searches for the global optimum during the stochastic optimization. Greedy algorithm converges faster than the GA which stems from the fact that once it discovers the local optimum, the search process is stop. Green BS deployment achieves almost the same performance as greedy algorithm, however, it needs 4 BSs because BS is added one-by-one without coordination. Besides, over the multiple runs of these algorithms, greedy algorithm attains the consistent results while the GA may have different outputs, because we fix the initial solution and search direction of greedy algorithm. The proposed algorithms achieve better coverage performance with less BSs than the actual

TABLE 5. Performance comparison under different simulation thresholds of GA.

Number	Threshold ϕ (dBm)	C_{th}	Running time (s)	Number of BSs	Coverage rate	Optimization Logs (number of BSs: coverage rate)
1	-90	70%	553.06	3	0.78	2:0.61, 3:0.78
2	-90	80%	1132.3	4	0.85	2:0.61, 3:0.78, 4:0.85
3	-90	90%	4037.6	6	0.95	4:0.85, 5:0.89, 6:0.95
4	-92	90%	956.73	4	0.92	2:0.65, 3:0.84, 4:0.92
5	-95	90%	308.48	2	0.94	2:0.94

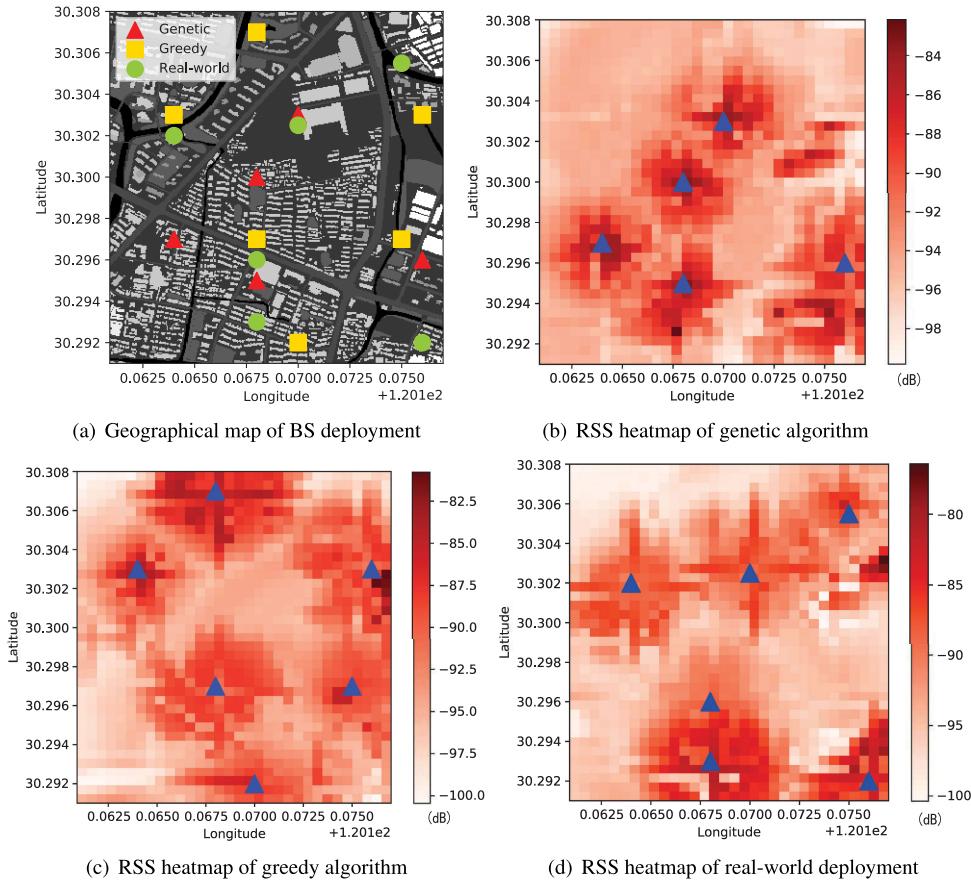


FIGURE 7. An example of BSs deployment results for two algorithms and the actual deployment, including the map of BS deployment and the heatmap of RSS (dBm).

deployment, which shows about 7.68% gain by greedy algorithm and 18.5% gain by GA in terms of coverage rate.

In Fig. 5, the effects on coverage performance, the number of BSs deployed and running time for different map resolution are simulated. The original resolution of map is 20m, which means that the region is divided into many square pixels with size of 20m×20m. However, in each iteration of online evaluation, the information of every grid has to be collected for RSS prediction. Therefore, the time consumption is highly related to map resolution since the lower is the map resolution, the larger is the number of grids and the measure space of feasible solution. Note that the map resolution has a slightly effect on the deployment since the coverage rate and the number of deployed BSs are almost the same, therefore, resolution of 100m is used when the deployment area is over 10 km² for fast convergence and resolution of 20m is used when the region is small, for example, 1 km².

Fig. 6 presents the results of execution time of the genetic algorithm and greedy algorithm under different area and number of deployed BSs. As expected, the convergence time of the two algorithms increases with the increment of planning area and number of deployed BSs, since they require more time to collect data for RSS prediction. The execution time of greedy algorithm grows linearly as the planning area and the number of deployed BSs increase, because the time mainly comes from the data collection brought by the increased BS number while the total number of iterations is almost the same. In contrast, genetic algorithm takes explorations until it reaches the global optimum or the maximum number of iterations, which results to exponential growth.

Table 5 gives the comparison of average program running results under different simulation thresholds in GA, where logs contains the record results during program optimization including the number of deployed BSs and the corresponding

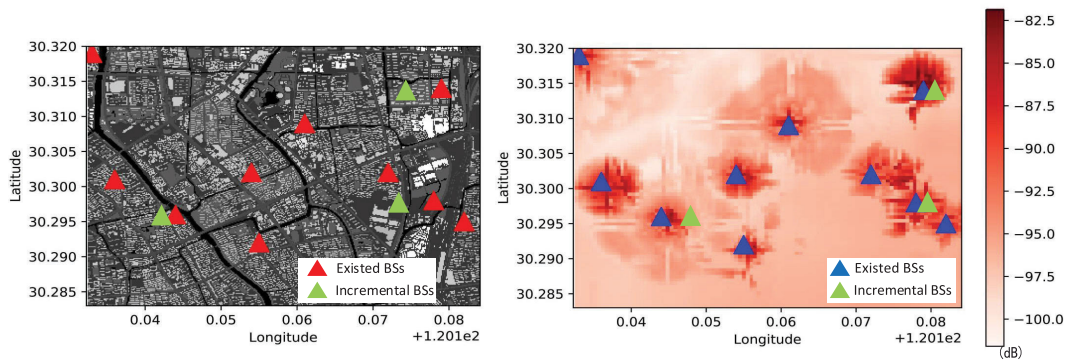


FIGURE 8. An example of incremental BSs deployment scenario.

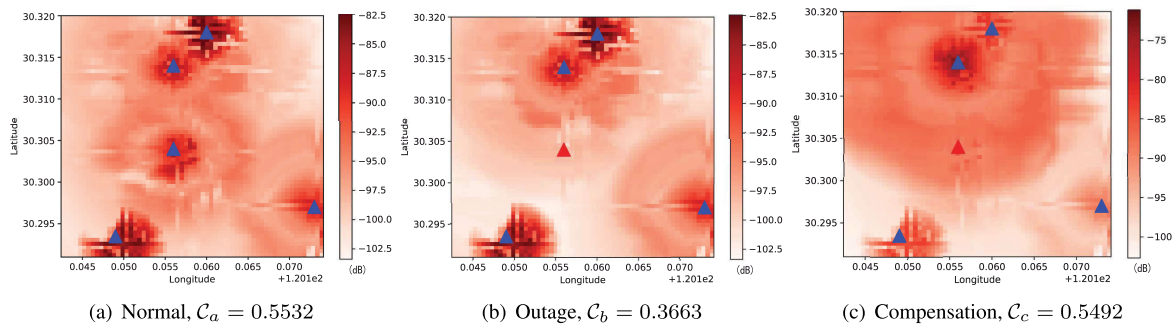


FIGURE 9. A use case that neighbor BSs react to fill the coverage gap caused by the faulty BS, where the blue triangle denotes the normal BS and the red triangle denotes the faulty BS. $\phi = 90$ dB.

optimal coverage rate. As expected, the number of BSs that need to meet the coverage threshold C_{th} is increasing according to the increment of C_{th} , because more BSs can provide higher quality coverage. Besides, if we lower the value of ϕ , fewer BSs are needed to be deployed because more RSS is greater than ϕ . Note that with the increase of ϕ and C_{th} , more BSs need to be deployed to satisfy the requirements, which results to larger running time. Note that the running time of program is strongly related to the number of deployed BSs, which is due to the data collection during the online evaluation.

Fig. 7 gives an example of results in BS deployment for two algorithms and the actual deployments in a 3×3 km² with map resolution of 50 m. Fig. 7 (a) presents the BS locations in the real-world map. In order to obtain an overview of BS deployment performance, we create the heatmap of RSS distribution as is depicted in Fig. 7 (b)-(d), which consists of about $[60 \times 60]$ matrix of 3,600 values that represent the RSS with respect to the cell that has the strongest signal at each grid. Each grid corresponds to one pixel of 50m by 50m. The ϕ is set to -97 dBm, and the C_{th} is set to 90%. Note that the genetic algorithm satisfies the requirement with only 5 BSs deployed, while the greedy algorithm as well as real-world deployment has to deploy 6 BSs to achieve this goal. However, from the distribution of BSs, we can see that the greedy algorithm is more reasonable than

real-world deployment since the coverage rate is 0.9628 versus 0.9189.

Fig. 8 gives an example of incremental scenario of BSs deployment. There have already existed 7 BSs in a 4×5 km² area and needs to deployment some BSs for better coverage. Greedy algorithm is leveraged to optimize the coverage rate, and we set the C_{th} to 70%. Before deployment, the coverage rate is 0.618 with ϕ of -97 dBm; after deploying 3 more BSs, the coverage rate is up to 0.703. Note that although the solution is generated randomly, which means that there are many solutions to satisfy the coverage requirement, the initial feasible solution and search direction have been fixed so that it generates the same result no matter how many times to run the program.

A use case of BS outage compensation is presented in Fig. 9. In Fig. 9 (a), the RSS heatmap of normal network operation is shown. If the middle BS is down, it will cause coverage gap and the coverage rate falls to 0.3663 as is shown in Fig. 9 (b). The tuned configurations consists of the antenna tilt, transmit power and azimuths of the neighboring BSs. Surrounding BSs automatically adjust their parameters according to the result generated by the algorithms until the coverage gap is filled, which is shown in Fig. 9 (c). Therefore, the network planning tool is applicable for practical scenarios of incremental deployment and BS outage compensation.

V. CONCLUSION

In this paper, a network planning tool that integrates machine learning techniques and heuristic algorithms, is proposed for RSS prediction and BS deployment. The goal is to meet the coverage requirement while minimizing the number of deployed BSs. In order to achieve this target, we divided the task into two stages. Firstly, a large amount of relevant information in real-world is fed into data processing module. Then, we train a regression model to predict RSS values. In stage II, the best parameters of multiple BSs is determined by leveraging the trained model and online optimization algorithms. Moreover, typical scenarios including increment deployment and BS outage compensation are investigated for practical needs. Numerical results demonstrate the RSS prediction model outperforms existing methods and the proposed BS deployment algorithms achieve better coverage than real-world networks with less BSs, which is applicable for practical systems.

REFERENCES

- [1] H. Zhang, Y. Chen, Z. Yang, and X. Zhang, "Flexible coverage for backhaul-limited ultradense heterogeneous networks: Throughput analysis and η -optimal biasing," *IEEE Trans. Veh. Technol.*, vol. 67, no. 5, pp. 4161–4172, May 2018.
- [2] H. Zhang, Z. Yang, Y. Liu, and X. Zhang, "Power control for 5G user-centric network: Performance analysis and design insight," *IEEE Access*, vol. 4, pp. 7347–7355, 2016.
- [3] N. Banerjee, M. D. Corner, D. Towsley, and B. N. Levine, "Relays, base stations, and meshes: Enhancing mobile networks with infrastructure," in *Proc. 14th ACM Int. Conf. Mobile Comput. Netw.*, 2008, pp. 81–91.
- [4] M. Hata, "Empirical formula for propagation loss in land mobile radio services," *IEEE Trans. Veh. Technol.*, vol. 29, no. 3, pp. 317–325, Aug. 1980.
- [5] V. Erceg, L. J. Greenstein, S. Y. Tjandra, S. R. Parkoff, A. Gupta, B. Kulic, A. A. Julius, and R. Bianchi, "An empirically based path loss model for wireless channels in suburban environments," *IEEE J. Sel. Areas Commun.*, vol. 17, no. 7, pp. 1205–1211, Jul. 1999.
- [6] S. K. Yoo, S. L. Cotton, R. W. Heath, Jr., and Y. J. Chun, "Measurements of the 60 GHz UE to eNB channel for small cell deployments," *IEEE Wireless Commun. Lett.*, vol. 6, no. 2, pp. 178–181, Apr. 2017.
- [7] CelPlan. (2019). *Celtrace Tm Wireless Global Solutions*. [Online]. Available: <http://www.celplan.com/products/indoor/celtrace.asp>
- [8] S. Hamalainen, H. Sanneck, and C. Sartori, *LTE Self-Organising Networks (SON): Network Management Automation for Operational Efficiency*. Hoboken, NJ, USA: Wiley, 2012, pp. 13–38.
- [9] N. Faruk, A. Ayeni, and Y. A. Adediran, "On the study of empirical path loss models for accurate prediction of TV signal for secondary users," *Prog. Electromagn. Res. B*, vol. 49, pp. 155–176, 2013.
- [10] J. Chen, U. Yatnalli, and D. Gesbert, "Learning radio maps for UAV-aided wireless networks: A segmented regression approach," in *Proc. IEEE Int. Conf. Commun. (ICC)*, May 2017, pp. 1–6.
- [11] S. Navabi, C. Wang, O. Y. Bursalioglu, and H. Papadopoulos, "Predicting wireless channel features using neural networks," in *Proc. IEEE Int. Conf. Commun. (ICC)*, May 2018, pp. 1–6.
- [12] S. I. Popoola, E. Adetiba, A. A. Atayero, N. Faruk, and C. T. Calafate, "Optimal model for path loss predictions using feed-forward neural networks," *Cogent Eng.*, vol. 5, no. 1, pp. 1–18, 2018.
- [13] J. Moysen, L. Giupponi, and J. Manges-Bafalluy, "On the potential of ensemble regression techniques for future mobile network planning," in *Proc. IEEE Symp. Comput. Commun. (ISCC)*, Jun. 2016, pp. 477–483.
- [14] L. Dai, H. Zhang, and Y. Zhuang, "Propagation-model-free coverage evaluation via machine learning for future 5G networks," in *Proc. IEEE 29th Annu. Int. Symp. Pers., Indoor Mobile Radio Commun. (PIMRC)*, Sep. 2018, pp. 1–5.
- [15] X. Li, D. Guo, J. Grosspietsch, H. Yin, and G. Wei, "Maximizing mobile coverage via optimal deployment of base stations and relays," *IEEE Trans. Veh. Technol.*, vol. 65, no. 7, pp. 5060–5072, Jul. 2016.
- [16] S. Chatterjee, M. J. Abdel-Rahman, and A. B. MacKenzie, "Optimal base station deployment with downlink rate coverage probability constraint," *IEEE Wireless Commun. Lett.*, vol. 7, no. 3, pp. 340–343, Jun. 2018.
- [17] G. Prasad, D. Mishra, and A. Hossain, "Coverage-constrained base station deployment and power allocation for operational cost minimization," in *Proc. IEEE 28th Annu. Int. Symp. Pers., Indoor, Mobile Radio Commun. (PIMRC)*, Oct. 2017, pp. 1–5.
- [18] F. Chernogorov and J. Puttonen, "User satisfaction classification for minimization of drive tests QoS verification," in *Proc. IEEE 24th Annu. Int. Symp. Pers., Indoor, Mobile Radio Commun. (PIMRC)*, Sep. 2013, pp. 2165–2169.
- [19] M. A. Yigitel, O. D. Incel, and C. Ersoy, "Dynamic BS topology management for green next generation HetNets: An urban case study," *IEEE J. Sel. Areas Commun.*, vol. 34, no. 12, pp. 3482–3498, Dec. 2016.
- [20] X. Sui, H. Zhang, and Y. Lv, "Coverage performance analysis of grid distribution in heterogeneous network," in *Proc. IEEE 17th Int. Conf. Commun. Technol. (ICCT)*, Oct. 2017, pp. 1424–1428.
- [21] F. Pedregosa, "Scikit-learn: Machine learning in Python," *J. Mach. Learn. Res.*, vol. 12 no. 10, pp. 2825–2830, 2012.
- [22] G. E. A. P. A. Batista and M. C. Monard, "An analysis of four missing data treatment methods for supervised learning," *Appl. Artif. Intell.*, vol. 17, nos. 5–6, pp. 519–533, May 2003.
- [23] S. Roweis, "EM algorithms for PCA and SPCA," in *Proc. Adv. Neural Inf. Process. Syst.*, 1998, pp. 626–632.
- [24] A. Shukla, H. M. Pandey, and D. Mehrotra, "Comparative review of selection techniques in genetic algorithm," in *Proc. Int. Conf. Futuristic Trends Comput. Anal. Knowl. Manage. (ABLAZE)*, Feb. 2015, pp. 515–519.



LINGCHENG DAI (Graduate Student Member, IEEE) received the bachelor's degree in communication engineering from the Beijing University of Posts and Telecommunications (BUPT), in 2017, where he is currently pursuing the M.Tech. degree in communication and information engineering with the School of Information and Communication Engineering. His research interest includes the emerging technologies of 5G wireless communication network.



HONGTAO ZHANG (Senior Member, IEEE) received the Ph.D. degree in communication and information systems from the Beijing University of Posts and Telecommunications (BUPT), China, in 2008. He is currently a Full Professor with the School of Information and Communications Engineering, BUPT. He has published over 100 articles on international journals and conferences, and has filed more than 50 patents. He is the author of ten technical books. His research interests include 5G wireless communication and signal processing.