IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received January 14, 2019, accepted February 5, 2019, date of publication February 15, 2019, date of current version March 8, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2899736

Comparative Study of Supervised Learning and Metaheuristic Algorithms for the Development of Bluetooth-Based Indoor Localization Mechanisms

JESÚS LOVÓN-MELGAREJO[1], MANUEL CASTILLO-CARA[1], OSCAR HUARCAYA-CANAL[1], LUIS OROZCO-BARBOSA[2], (Member, IEEE), AND ISMAEL GARCÍA-VAREA[2]

[1]Center of Information and Communication Technologies, Universidad Nacional de Ingeniería, Lima 25, Peru
[2]Research Institute of Informatics, Universidad de Castilla-La Mancha, 02071 Albacete, Spain

Corresponding author: Manuel Castillo-Cara (mcastillo@uni.edu.pe)

This work has been partially funded by the "Programa Nacional de Innovación para la Competitividad y Productividad, Innóvate - Perú" of the Peruvian government, under Grant No. 363-PNICP-PIAP-2014, by the Spanish Ministry of Economy and Competitiveness under Grant numbers TIN2015-66972-C5-2-R and TIN2015-65686-C5-3-R, and by the Regional Council of Education, Culture and Sports of Castilla-La Mancha under grant number SBPLY/17/180501/000493, supported with FEDER funds.

ABSTRACT The development of the Internet of Things (IoT) benefits from 1) the connections between devices equipped with multiple sensors; 2) wireless networks and; 3) processing and analysis of the gathered data. The growing interest in the use of IoT technologies has led to the development of numerous diverse applications, many of which are based on the knowledge of the end user's location and profile. This paper investigates the characterization of Bluetooth signals behavior using 12 different supervised learning algorithms as a first step toward the development of fingerprint-based localization mechanisms. We then explore the use of metaheuristics to determine the best radio power transmission setting evaluated in terms of accuracy and mean error of the localization mechanism. We further tune-up the supervised algorithm hyperparameters. A comparative evaluation of the 12 supervised learning and two metaheuristics algorithms under two different system parameter settings provide valuable insights into the use and capabilities of the various algorithms on the development of indoor localization mechanisms.

INDEX TERMS Indoor positioning, fingerprinting, Bluetooth, classification model, signal processing, received signal strength indication, multipath fading, transmission power, benchmark, metaheuristic optimization algorithms.

I. INTRODUCTION

A number of wireless network technologies are currently available in the market, of which Wi-Fi and Bluetooth are by far the most popular. This is because most current smartphones have Wi-Fi and Bluetooth interfaces. Accordingly, most research and development efforts in the area of wireless indoor localization mechanisms have been made using one or both of these wireless technologies [1]. As for other technologies, Zigbee has also been explored in the context of wireless sensor networks [2], [3]. These studies are being conducted using the received signal strength indication (RSSI) of various wireless transmitters as a means of estimating the location of a smartphone device [4]. Among the technologies being considered, over the past years, Wi-Fi networks have attracted the attention of many researchers and practitioners who have employed innovative techniques, e.g., machine and deep learning [5]. Many experimental studies have been conducted to construct radio maps and models as a means to estimate the distance between a reference transmitter and a smartphone device. Because of the characteristics of the wireless signal, the use of Kalman filters [6], [7] has been required to remove the noise. Novel Bluetooth Low-Energy 4.0 (BLE4.0) devices have become a strong alternative to Wi-Fi-based indoor location mechanisms. Their low cost, low energy consumption, and the size of the Bluetooth devices are among the most important design features of battery-operated mobile devices, mainly smartphones and tablets. Moreover, these devices have many sensors, e.g., accelerometer, that can be used to assist the indoor localization process [8]. In this context, Table 1 lists the main characteristics of these three wireless technologies. In the case of Bluetooth, the table lists the BLE4.0 specifications. The table also includes the main algorithmic techniques used in the characterization of RSSI fingerprints generated by the wireless devices [9], [10].

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

TABLE 1. Comparison of Bluetooth, Wi-Fi and ZigBee technologies.

	Bluetooth	Wi-Fi	ZigBee
Frequency Band	2.4GHz	2.4/5GHz	868MHz (EU) 915MHz (USA) 2.4GHZ
Data Rate	1Mbps	11/54Mbps	250Kbps
Range	10 to 100m	Up to 100m	Up to 20Km (depend of the frequency)
Power Consump.	Very low	High	Very low
Battery Life	Multiple months	Multiple hours	Multiple months
Infrastructure	To be deployed	Existing Wi-Fi nodes	To be deployed
Smartphones	Supported	Supported	Not supported
Main algorithmic technique	Supervised Learning Algorithms.	Probabilistic graphical models.	Probabilistic graphical models.
Typical applications	Sensors, positioning, peripherals.	WLAN, broadband, connections.	Industrial control and monitoring, sensor networks.

Wi-Fi and Zigbee were primarily designed for implementing wireless communications LANs, including broadband connections, and the deployment of distributed wireless monitoring and actuator applications, respectively. Due to a large number of hotspots based on Wi-Fi access points and the deployment of Zigbee wireless sensor networks, developers and practitioners are exploring the development of localization and tracking mechanisms based on these two wireless technologies.

Among the different approaches being pursued in the development of localization mechanisms, those approaches based on the characterization of RSSI fingerprints have benefited from the use of probabilistic graphical models (PGMs) [7], [11] and supervised learning algorithms (SLAs) [12]. The localization and tracking of a target within a given area is possible based on the characterization of the RSSI fingerprints generated by a set of wireless transmitters [6], [9].

Therefore, the methodology to develop a localization mechanism based on RSSI fingerprinting using a SLA consists of two main phases: (i) characterization of the distribution of the RSSI in the area enabling the localization of a given target [4], [13]; and (ii) the evaluation of the accuracy and error of the localization mechanism [2]. Brunato and Battiti [14] presented a set of SLA techniques applied to Wi-Fi fingerprinting, and a benchmark to compare them, obtaining good results.

Considering the latest developments and technologies, i.e., BLE4.0 beacons, hereinafter referred to as beacons, this work is an extension of our previous research efforts [12], [15]. The two major extensions are: the evaluation of twelve different SLAs: linear, non-linear, and ensemble models; and the use of genetic algorithms (GA) as a means to reduce the computational cost and the time to optimize the setting of the transmission power levels, hereinafter referred to as TxPower, of every transmitter. Our proposal involves the optimization of the hyperparameters of the various algorithms.

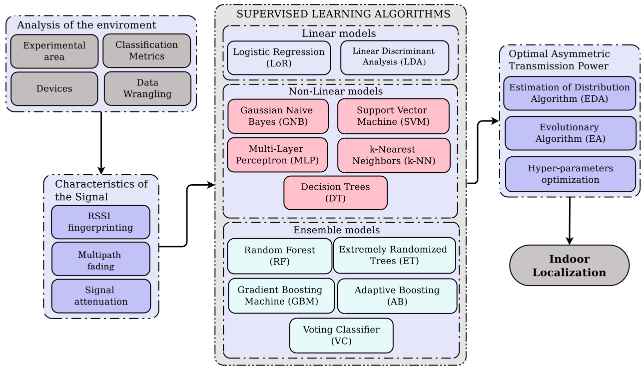


FIGURE 1. Overall schema proposal.

The remainder of this paper is organized as follows. Section II reviews the recent BLE4.0 localization literature, including the techniques followed in our works in the area of signal processing analysis. Section 3 specifies our indoor setting and the devices used as transmitters and receivers, depicted in Figure 1 with blocks called "Analysis of the Environment" and "Characteristic of the Signal." In addition, RSSI and its behavior, i.e., signal features, are discussed. Subsequently, Section 4 explains the SLAs used, whose performance will be evaluated according to the accuracy and mean error classification metrics, represented in Figure 1 with the block called "Supervised Learning Algorithms." Section 5 presents our first set of results using symmetric and asymmetric TxPower configurations of the beacons. Section 6 introduces and evaluates the performance of the metaheuristic algorithms and the tuning of the hyperparameter used on the search of the optimal asymmetric TxPower configuration. This section also includes an analysis of the computational cost of the twelve SLAs considered in our study, depicted in Figure 1 as "Optimal Asymmetric Transmission Power." Finally, Section 7 presents our conclusions and future work directions.

II. RELATED WORK

In this section, we introduce the main features of the BLE4.0 technology and its relevance on setting indoor localization facilities. For better understanding, this section has been divided into two subsections, which explain the technical characteristics, limitations, and communication protocol of BLE4.0 related to indoor localization techniques.

A. BLE4.0 SIGNAL PROCESSING

BLE4.0 technology has rapidly spread in recent years. It is available in most mobile devices, such as smartphones, tablets [16], and electronic development kits [17]. Beacons emit short packets, characterized by providing ways of determining zones of proximity through the intensity of the signal, i.e., RSSI. Beacons have low power consumption requirements, making it possible for them to operate for long periods without the need of replacing their batteries.

BLE4.0 divides the band into 40 2-MHz channels. Since Wi-Fi and BLE4.0 operate over the 2.4-MHz band, beacons make use of channels 37 (2402-MHz), 38 (2426-MHz) and 39 (2480-MHz) to avoid interference between devices and advertise their presence [18]. Beacons cyclically broadcast on these channels and use the other channels once paired with a BLE4.0-equipped device. Beacons can transmit signal in increments from 100ms to 10.24s, in steps of 0.625ms.

This parameter directly affects the battery lifetime. The beacons have configurable TxPower levels that usually range from −30dBm to 4dBm. The signal strength conditions the beacons sensing range [17]. Therefore, the distribution of the RSSI spectrum in the area depends directly with the TxPower level [12], [15]. BLE4.0 signals are prone to noise and impairments due to the presence of physical elements within the coverage area, such as furniture, people, walls, windows, and other obstacles. This makes it necessary to conduct RSSI surveys. Some works have reported readings of a given location with signal level variations of up to 20dB in less than 20s [9], [10]. The deployment of beacons must be carefully planned. In particular, the placement and all relevant system BLE4.0 parameters must be calibrated to meet the end-user expectations [19].

Multipath fading (MPF) is another major impairment that has a major impact on the design of indoor wireless localization mechanisms. Recent results have shown that the use of floor plan as a basis for identifying the multipath components may be exploited to enhance the accuracy of wireless indoor localization schemes [9], [17], [20]. Although the use of such schemes is still in its infancy and limited to wide-band communications, insights into the impact of the structural features on the RSSI metric have been obtained. In previous research [9], Faragher and Harle applied the multipath mitigation algorithm to the RSSI fingerprint of their BLE4.0 experimental setup.

Various other works have explored the use of different TxPower and channels to identify the setup offering the best results. In a previous study [10], the authors analyze the impact of the channel used for collecting the RSSI samples, which revealed major differences in the level of the signal samples. Another research [21] reported that power plays a major role in terms of system performance, and its results demonstrated the benefits of using the *k*-Nearest Neighbors (*k*-NN) algorithm as a means to better exploit the information retrieved from the RSSI fingerprint in the development of indoor wireless localization schemes.

B. BLE4.0 INDOOR LOCALIZATION ALGORITHMS

Depending on the wireless network technology, the use of a specific technique/algorithm may be more suitable or feasible than the use of other techniques. As already stated, most works to date reported that noise and MPF are two of the main impairments with a negative impact on the quality of RSSI fingerprints. Improving or identifying the impact of such impairments on the quality of the RSSI fingerprint is therefore one of the main challenges in the development of robust and accurate BLE4.0-based indoor localization mechanisms [4]. Since the structural characteristics and the layout of objects may play a major role in signal impairments, the research community is actively working on defining the best system configuration, e.g., density of beacons and relative placement [22] and identifying the most suitable data processing methodologies, i.e., filtering and classification algorithms.

Recent studies employing SLAs have reported promising results on characterizing RSSI fingerprints in the presence of noise and MPF effects. In previous research [14], various SLAs that were applied to Wi-Fi fingerprints were compared. In a more recent work [23], a hybrid localization experiment was conducted using a set of Wi-Fi access points (APs) accompanied by BLE4.0 devices. The localization mechanism was based on weighted nearest neighbors in the signal space algorithm. The main objective of this study was to improve the indoor location by using BLE4.0 devices and deploying a system that is constantly updated according to RSSI levels reported by mobile devices (receivers). During the experiments, two parameters of the BLE4.0 devices were varied: the scan duration of the RSSI signal and the density. In contrast, the TxPower was fixed to the maximum level throughout the experiments.

In [14], in addition to exploring and performing an extended evaluation of two SLAs, support vector machine (SVM) and *k*-NN, the authors also explored the use of Bayesian modeling and multilayer perceptron (MLP) algorithms. Although their work is developed for Wi-Fi, the authors claimed that their regression and classification algorithms can be applied to the analysis of RSSI fingerprints created by other wireless technologies.

While most recent works consider *k*-NN and SVM as the two most promising SLAs, other works are exploring the use of deep learning techniques as another alternative for improving the quality of the information extracted from RSSI fingerprints that consider two main metrics: accuracy, and performance [24].

The above review has proven useful for defining and guiding the objectives and methodology of our research. After identifying the major system parameters and impairments, the experimental setup layout was defined. The following main system parameters were identified: number of beacons, TxPower, and advertisement period. In one of our previous works, we have shown that the use of SLA algorithms may prove beneficial on mitigating the MPF impairment [15]. The optimal parameter setting, namely the transmission power setting of the Bluetooth beacons was conducted using a brute force approach. In this work, we go a step further by exploring twelve SLA algorithms following three different paradigms. We also explore the use of two metaheuristic approaches as a means to reduce the computational requirements. Our work the differs from previous works that have mainly focused on a limited number of SLA algorithms without taking into account their underlying features. Furthermore, to the authors knowledge, no previous works have made use of metaheuristic algorithms as a means of reducing the computational requirements on the process of setting the best system configuration.

III. BACKGROUND: TOOLS AND WIRELESS SIGNAL CHARACTERIZATION

The design and development of BLE4.0 fingerprint localization techniques present major challenges since the indoor propagation of BLE4.0 signals is highly sensitive to the MPF effect [25]. It is also widely recognized that the capabilities of the surveying devices will play a major role in the quantity, quality, and time of the effort invested to produce valuable RSSI fingerprints. The details about the area, transmitter/receiver, survey campaigns, MPF, and intraday signal attenuation were analyzed in previous works. Hence, we discuss additional information about BLE4.0 signal characterization and conduct an in-depth analysis about the impact of different materials/structures on RSSI [26], [27].

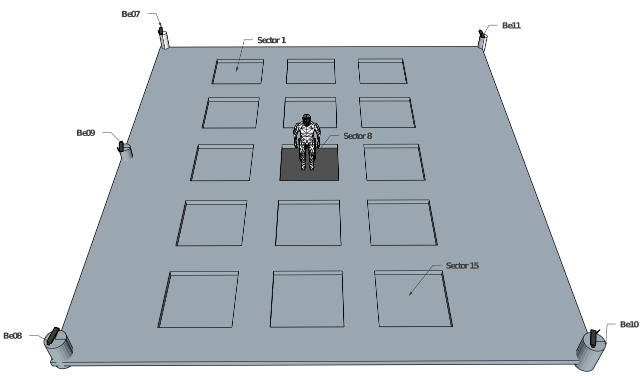
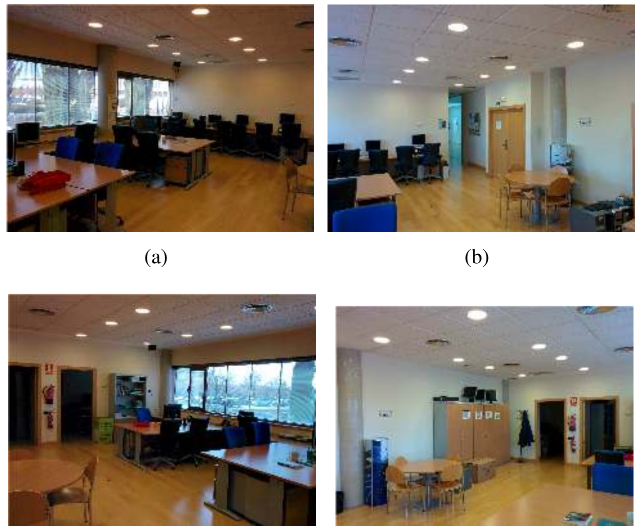


FIGURE 2. Beacon indoor experimental area setup.

A. EXPERIMENTAL AREA

Our experiments were conducted in the lab of our research institute. We placed four beacons at each one of the four corners of a 9.3m $\times$ 6.3m rectangular area. The fifth beacon was placed in the middle of one of the longest edges of the room. Figure 2 depicts the experimental area in which the five Beacons have been labeled as Be07, Be08, Be09, Be10, and Be11. We divided the experimental area into 15 sectors of 1m^2, each separated by a guard distance of 0.5m. A 1.5m-wide strip was left around the experimental area. This arrangement will allow us to better differentiate the RSSI level of joint sectors when reporting our results. Measurements were performed by placing the mobile device at the center of each one of the 15 sectors, as described below. The shortest distance between a beacon and a receiver was limited to 1.5m. Figure 3 shows four views taken from each one of the four corners of the lab. As shown in the figure, we placed beacons Be10 and Be11 in front of a window, Figures 3(d) and 3(b), respectively, while all other beacons were placed in front of the opposite plasterboard wall. We further noticed that beacon Be08 was placed at the left edge of the entrance door, close to the corridor with a glass wall (Figure 3(c)).



(a) (b) (c) (d)

FIGURE 3. Pictures from each one of the four corners of the lab. (a) View from Be07. (b) View from Be11. (c) View from Be08. (d) View from Be10.

B. TRANSMITTER AND RECEIVER DEVICES

For this experiment, JAALEE beacon devices were used [28]. According to the specifications of the five Beacons used in our experiments, they may operate at one of eight different TxPower levels. Following the specifications, the TxPower levels are labeled in consecutive order from the highest to the lowest level as TxPower $= 0\times01$, TxPower $= 0\times02, \ldots,$ TxPower $= 0\times08$ (ultra wide range transmission power: 4dBm to -40dBm), although TxPower $= 0\times07$ and TxPower $= 0\times08$ were discarded since they did not adequately cover the signal spectrum in the entire area. During our experiments, we conducted several measurement campaigns by fixing the TxPower level of all beacons at the beginning of each campaign. Furthermore, all measurements were performed under line-of-sight conditions.

As a receiver, we used a Raspberry Pi equipped with a USB BLE4.0 antenna [29], hereinafter referred to as BLE4.0 antenna.

C. RSSI FINGERPRINTINGS

It is well known that the floor plan and materials are two major parameters that have a significant impact on the indoor propagation of BLE4.0 signals. Since the experiments were conducted in a single lab and zero occupancy during most trials, the experiment focused on the impact of the surrounding wall materials.

As shown in Figure 2, we analyzed the four beacons placed at the corners with TxPower $= 0\times04$ taking the RSSI in all sectors. Beacon Be09 and other TxPower levels were omitted because they had similar output and did not add extra information. Figure 4 depicts the RSSI measurements

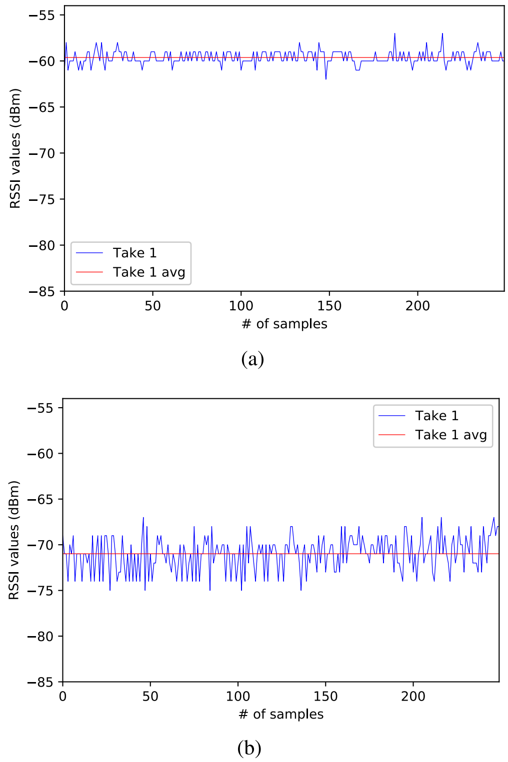


FIGURE 4. RSSI behavior using TxPower = 0×04. (a) Sector 13 for Be08. (b) Sector 15 for Be10.

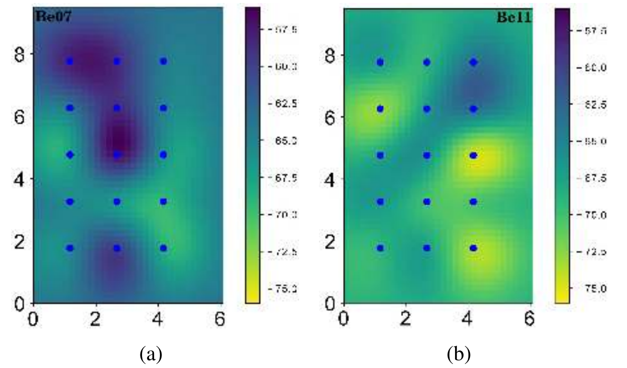


FIGURE 5. RSSI fingerprinting for different beacons using TxPower = 0×04 for: (a) Be07 and (b) Be11. Axis X and Y represent the size (in meters) of the experimental area.

taken at Sectors 13 and 15, corresponding to the signals of Be08 and Be10, respectively. Be10 is located at a corner of a flat wall made of drywall and a window wall located at the right side of Figure 3. Be08 is placed at the corner of a flat wall made of drywall and at the entrance of a corridor. Comparing the RSSI levels captured for Be08 (Figure 4(a)) to those captured for Be10 (Figure 4(b), the signal of Be10 suffers higher attenuation and less RSSI value, a difference of more than 10dBm, than that experienced for the signal of Be08. These results clearly show the challenges involved in accurately characterizing the indoor signal propagation of a BLE4.0 transmitter.

To gain further insight into the impact of the surrounding material on indoor signal propagation, Figure 5 shows the RSSI fingerprints throughout the experimental area of Be07 and Be11. In this case, Be11 is located at the corner of a flat wall made of drywall and a window wall located at the right-hand side in Figure 2. Be07 is placed at the corner of a flat wall that is solely made of drywall. In this case, we can see the same behavior of the RSSI which Be07 has a greater intensity than Be11.

From the results shown in Figures 5 and Figure 5, it is clear that the materials surrounding BLE4.0 have a major impact on signal propagation. The main observations can be summarized as follows:

- The intensity of the RSSI in sectors close to the window side is lower than that at the sectors close to the drywall side.
- The RSSI levels of the beacons placed at the windows, namely, Be10 and Be11, experience a higher attenuation than the beacons placed at the walls made of drywall, namely Be07 and Be08.

Furthermore, the RSSI fingerprints (Figure 5), illustrate the challenges involved in characterizing signal propagation. It is evident that the presence of the MPF effect makes it difficult to estimate the distance between the receiver and transmitter based on the RSSI level.

These results clearly show that the setting of the TxPower levels may prove useful in overcoming the impact of the MPF effect. The mitigation of this effect will provide us with the means to improve the accuracy of the classification algorithms. This finding has motivated most results reported in our previous works [15].

IV. MACHINE LEARNING MODELS: PRINCIPLES AND EVALUATIONS

Machine learning techniques include a series of models in which systems can perform the learning process to make the best possible decisions or accurate predictions based on the information extracted from a large dataset. This study was extended to 12 SLAs, which were grouped as linear, non-linear, and ensemble models. Moreover, this section explains the main classification metrics and the data collected during the survey.

A. SUPERVISED LEARNING ALGORITHMS

For this work, SLAs, specifically classification models, prove useful in solving indoor location fingerprinting. Hence, in SLAs, we know the input parameters and the output with which we iteratively train the dataset. Once the model is trained, predictions are made and compared with ground truth values to obtain an estimation of the performance of the model.

The total analysis of this work has been conducted using different classification models (explained in Table 2) and divided into a taxonomy of three models:

TABLE 2. Definition of the different machine learning models evaluated.

Linear Models	Logistic Regression (LoR)		For binary classification problems. In this case, this model helps to determine whether the input belongs to a specific sector. It uses the sigmoid function that has a range of output values between 0 and 1.
	Linear Discriminant Analysis (LDA)		This model helps to find the maximum separation between data groups to classify them in classes. Its approach is similar to PCA, which reduces the dimensions of high-dimensional data via linear combinations, and it can be extended to multiclass classification problems.
Non-Linear Models	Gaussian Naive Bayes (GNB)		It is a binary and multiclass classification algorithm useful for large datasets. It performs classification assuming the independence between the features (naive) and calculates the classes according to the Bayesian probability.
	Multi-Layer Perceptron (MLP)		It uses the neuron structure, where the variable input is represented by the dendrites and the output is calculated via a nonlinear activation function, to first learn and then perform classification. This algorithm uses three types of neuron layers: input layer, hidden layer(s), and output layer.
	Support Vector Machine (SVM)		Given the data in the space, it builds hyperplanes in a high-dimensional space with a maximum gap between them. With the aid of kernel functions, it can perform the classification for high-dimensional data.
	k-Nearest Neighbors (k-NN)		This algorithm classifies the input based on a measure of similarity, which is often the distance in the space of the data points. A prediction is made by choosing the most frequent class between the k-NN.
	Decision Tree (DT)		A predictive model that places the observations made from the data in the branches; these lead to the leaves that are labeled with the correct classification. It uses a discrete set of values, and the leaves yield the final output.
Ensemble Models	Bootstrap Aggregation (Bagging)	Random Forest (RF)	Each tree of the forest is built from an extract of the data processing and random subset of features. As a result, the bias is slightly high in comparison with that observed in a non-random single tree. Overall, this is compensated with a low average variance.
		Extremely Randomized Trees (ET)	Like in RF one subset of features is used; however instead of using the most important, among them it is used as a random set. This analysis reduces the model variance more.
	Boosting	Adaptive Boosting (AB)	In each iteration, the weakest learners are used to improve the model according to the misclassified ones. A combination is made for every learner based on a weighted sum that yields the final answer of the boosted classifier.
		Gradient Boosting Machine (GBM)	This model makes predictions similar to those made by a set of weak learners, typically DTs. It builds a model in an optimal way to model like the boosting methods and generalizes the optimization function of arbitrary loss. It can handle data with atypical values in the output side.
	Voting	Voting Classifier (VC)	It is a simple way of combining the predictions of the GBM and ET.

- **Linear models**: In these models, we may expect the target value to be expressed as a linear combination of constant values or the product between a parameter and a predicting variable. In other words, the predicting variable may be modified, thereby generating more complex curves or shapes.
- **Non-linear models**: This models do not make strong assumptions about the relationship between the input attributes and the output attribute being predicted.
- **Ensemble models**: These models combine prediction models to improve the strength and the performance of the classification model. The three most popular ensemble models are:
  - **Bootstrap Aggregation** or **Bagging** involves taking multiple samples of the training dataset (with substitution) and forming a model for each sample. The prediction of the final output is averaged through the predictions of all sub-models.
  - **Boosting** algorithms create a sequence of models that attempt to correct the errors of the previous models in the chain. Once created, these models make predictions that can be weighted by their demonstrated accuracy and the results are combined to generate an output.
  - **Voting** is the linear combination of different classifiers weighted with different probabilities values in order to better predict the class labels.

B. CLASSIFICATION METRICS ASSESSMENT

Before the training phase, RSSI measurements are obtained by placing the receiver at different sectors. The RSSI measurements are then stored in a database during an offline phase, including the $\langle x, y \rangle$ coordinates and RSSI level for each sample. Afterward, the RSSI receiver measures are captured again in an online phase. These last instances are then compared with the derived model to predict the location of the receiver, i.e., to generate RSSI-based location fingerprinting. The selected area with the beacon position can be seen in Figure 2.

In this context, we have used two metrics related to the classification models:

- **Accuracy**: It is the most intuitive performance measure and it is simply a ratio of correctly predicted observation to the total observations. The value is calculated in percentage (%) for the whole experimental area. It describes the global accuracy rate of each setup.
- **Mean error**: The average error for the entire experimental area. This error is calculated in meters (m)

considering the total dimensions of each area, and it is computed by averaging all absolute differences between the predicted localization and the central point of each sector of the experimental area, that is the real position where each measurement was taken.

Moreover, for this research, the Scikit-learn Python library [30] was used. In all the experiments we used a 10-fold cross-validation scheme due to the size of the experimental data, and the low variance obtained in terms of accuracy and mean error compared other cross-validation fold values.

Finally, herein, data preprocessing was not performed. In other words, raw data or data as collected have been used for all experiments carried out. Table 3 lists the total number of datasets for each TxPower.

TABLE 3. Sample sizes of the RSSI captured using BLE4.0 at different transmission power (TxPower) levels.

Transmission Power	Sample Size per Beacon
TxPower=0x01	5004
TxPower=0x02	5246
TxPower=0x03	4844
TxPower=0x04	5134
TxPower=0x04	4697
TxPower=0x06	4198

V. EXPERIMENTAL RESULTS

In this section we present the results obtained in the experiments that we have carried out. First, we present a comparison of localization performance using symmetric and asymmetric TxPower setups. Here, the results obtained with a symmetric TxPower setup are considered as the baseline results.

A. BASELINE RESULTS: SYMMETRIC TRANSMISSION POWER SETUPS

In Table 4 the localization accuracy obtained for each symmetric TxPower configuration is presented for every classification models tested. As can be seen, linear models present the worst performance results. On the contrary, ensemble models have better accuracy, very similar to that of non-linear models, except for MLP, of which k-NN provides the best results. In addition, we can see that the GBM model obtains the best accuracy, followed by VC. With respect to the TxPower, using a TxPower $= 0\times05$ give the worst results for all classification models, followed by TxPower $= 0\times02$ and TxPower $= 0\times01$. In this context, we can see that for TxPower $= 0\times06$, the best results are obtained with GBM with an accuracy of 86.10% and the worst results are obtained with MLP with an accuracy of 55.40%. Another aspect to take into account is the behavior in terms of variance, mean, and accuracy range. This analysis can be seen in Figure 6, which shows the accuracy for each TxPower level in a box-plot graphics for TxPower $= 0\times06$ and TxPower $= 0\times02$ (see Figures 6(b) and 6(a)). In this figure, the remaining TxPower has not been included for a better understanding. On performing an analysis in the entire TxPower group, we observe that the non-linear and ensemble models have a constant behavior in average as well as a minimum variance, especially ET.

TABLE 4. Accuracy (%) for each classification model. best values for each TxPower are highlighted in bold.

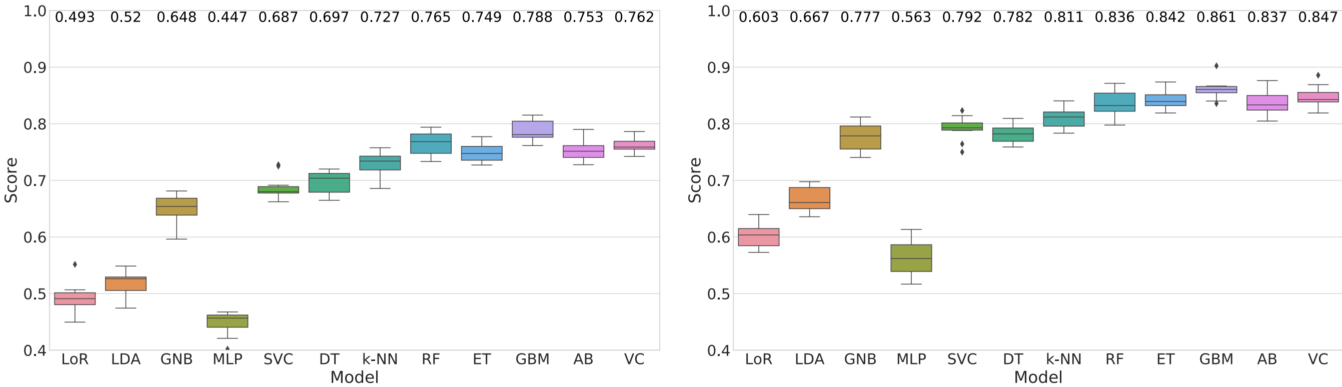
Model	0x01	0x02	0x03	0x04	0x05	0x06
LoR	60.50	49.30	50.80	57.20	48.90	60.30
LDA	63.80	52.00	57.90	62.60	54.30	66.70
GNB	73.90	64.80	72.20	73.60	65.00	77.70
MLP	56.00	45.30	44.00	53.20	43.80	55.40
SVM	72.20	68.70	74.90	74.10	67.20	79.20
DT	73.80	69.70	72.70	75.00	64.20	78.20
k-NN	77.50	72.70	77.00	75.70	70.00	81.10
RF	79.10	76.50	79.10	79.80	71.30	83.60
ET	78.50	74.90	79.20	78.70	71.30	84.20
GBM	**81.70**	**78.80**	**82.00**	**82.30**	**76.50**	**86.10**
AB	80.00	75.30	80.00	78.70	72.10	83.70
VC	79.40	76.20	80.00	80.10	73.80	84.70

Regarding the mean error, Table 5 lists the results obtained for each classification model and TxPower used. The results show that the ranking of the different algorithms is straightforward, i.e, the ensemble models exhibit the best results, followed by the non-linear models while the linear models report the worst results. This is true independently of the power transmission being used except for the case of the MLP algorithm. We also notice that the results reported by the different algorithms belonging to the non-linear and ensemble, differ by no more than one per cent, except once again in the case of the MLP algorithm. Similar conclusions can be drawn from the mean error results, see Table 5.

TABLE 5. Mean error (m) for each classification model. best values for each TxPower are highlighted in bold.

Model	0x01	0x02	0x03	0x04	0x05	0x06
LoR	0.933	1.270	1.433	1.111	1.605	1.104
LDA	0.869	1.162	1.309	0.937	1.387	0.912
GNB	0.644	0.835	0.885	0.674	0.991	0.649
MLP	1.078	1.190	1.650	1.191	1.765	1.131
SVM	0.772	0.766	0.783	0.753	1.056	0.636
DT	0.623	0.706	0.882	0.640	1.024	0.583
k-NN	0.555	0.702	0.660	0.594	0.958	0.525
RF	0.526	0.570	0.660	0.529	0.892	0.499
ET	0.516	0.585	0.647	0.525	0.927	0.490
GBM	**0.426**	**0.536**	**0.561**	**0.463**	**0.714**	**0.384**
AB	0.522	0.603	0.628	0.505	0.832	0.532
VC	0.502	0.592	0.597	0.491	0.852	0.490

Comparing Table 4 with 5 shows that in our setting the worst results in both accounts, accuracy and mean error are reported for the case when TxPower $= 0\times05$. However, more importantly, we can conclude from a more in-depth analysis of the results that a given power level does not guarantee the best results in both accounts. As far as the more appropriate algorithm, the GBM algorithm report the best results in terms of the two metrics of interest. In summary, to improve the accuracy and mean error, it is worth to explore the asymmetric transmission power setting of the beacons as a means to mitigate the multipath fading effect.



(a) (b)

FIGURE 6. Accuracy boxplots of classification models using symmetric TxPower setups: (a) TxPower = 0×02 and (b) TxPower = 0×06.

B. NUMERICAL RESULTS – ASYMMETRIC TRANSMISSION POWER SETUPS

Previously, it has been exposed the importance of using different TxPower levels for each beacon (hereinafter referred to as asymmetric TxPower configuration) to mitigate the MPF effect, which has an impact on the final localization results. Given this premise, five beacons with six TxPower levels (i.e. from TxPower = 0×01 to TxPower = 0×06) account for a total of 7776 combinations. In this section we will evaluate all these combinations to identify which configuration is the best for each classification model.

Table 6 summarizes the results of all Asymmetric TxPower configuration combinations evaluated. A specific TxPower configuration is represented as a vector that contains the TxPower level assigned to each beacon in this order: Be07, Be08, Be09, Be10, and Be11 (e.g. [6–1–3–3–5]). The table shows the accuracy (%) and mean error (m) results for each classification model using the best asymmetric TxPower configuration.

TABLE 6. Accuracy (%) and mean error (m) results for each classification model using asymmetric TxPower configuration. the values of the best classification model are highlighted in bold.

Model	TxPower	Acc	ME
LoR	[4–1–2–3–1]	70.85	0.647
LDA	[6–1–3–3–5]	77.18	0.486
GNB	[4–1–2–6–1]	86.01	0.289
MLP	[4–1–4–6–1]	68.18	0.692
SVM	[6–1–6–6–1]	83.49	0.460
DT	[6–1–3–6–5]	85.95	0.289
k-NN	[6–1–3–6–1]	87.78	0.264
RF	[6–1–3–3–5]	89.84	0.228
ET	[6–1–3–3–5]	89.25	0.197
GBM	**[6–1–3–3–5]**	**91.45**	**0.185**
AB	[6–1–3–6–5]	89.40	0.234
VC	[6–1–3–3–5]	90.43	0.200

Comparing these results with the ones obtained with symmetric TxPower configurations, that is results shown in Tables 4 and 5, we can see similar behaviors with respect the type of SLA used. In this respect, ensemble algorithms obtain the best results and the linear algorithms, except MLP, obtain the worst results. GBM continues to predominate in terms of accuracy and low mean error (91.45% and 0.185m, respectively), while MLP presents the worst results with an accuracy and mean error of 68.18% and 0.692m, respectively. As for the TxPower, except for AB which has a [6–1–3–3–5] configuration, we can observe how configuration [6–1–3–6–5] is the most repeated, the best configuration for ensemble algorithms. Regarding the linear and non-linear algorithms, we observe different optimal configurations.

In addition, we can observe common behavioral patterns in general lines for the different beacons:

- For Be07, TxPower = 0×06 predominates, with TxPower = 0×04 predominating to a relatively lesser extent.
- Be08 has TxPower = 0×01, the best TxPower level configured for all models.
- Be09 is has the greatest variation of TxPower (TxPower = 0×02, TxPower = 0×03, TxPower = 0×04 and TxPower = 0×06), although there is a majority of TxPower = 0×03.
- In Be10, we see equal alternation between TxPower = 0×03 and TxPower = 0×06 configurations.
- Be11 has an alternation between TxPower = 0×01 and TxPower = 0×05.

Regarding accuracy, as summarized in Table 6, there is a considerable improvement in the values obtained from each of the models. With these results and taking into account the symmetric TxPower results for TxPower = 0×06, we can see how the models have improved on average in terms of accuracy and mean error: (i) linear models improved by 10% and 0.500m; (ii) non-linear improved by 9% and 0.340m; and (iii) ensemble models improved by 5% and 0.260m.

However, following a brute-force search to find the optimal TxPower configuration is computationally unfeasible given that the search space (combinations of beacons and TxPower) grows exponentially with the number of beacons. Then, in order to search for the optimal asymmetric TxPower configuration we need to rely on metaheuristic search techniques. This issue is addressed in the next section in order to apply this techniques to real time indoor localization applications.

VI. OPTIMAL TXPOWER CONFIGURATION: METAHEURISTIC OPTIMIZATION AND PARAMETER TUNING

It is worth mentioning that the search for an optimal TxPower configuration requires high computational resources. Consequently, metaheuristic optimization algorithms may prove effective on computing the optimal asymmetric TxPower setup.

A. METAHEURISTIC ALGORITHMS

Here we discuss two metaheuristic approaches: (i) evolutionary algorithms (EAs), specifically genetic algorithms (GA); and (ii) estimation of distribution algorithm (EDAs).

Case 1 (Estimation of Distribution Algorithm): EDAs are stochastic optimization methods to search the optimum solution by building and sampling explicit probabilistic models of promising candidate solutions. In our case the algorithm randomly initializes the population (e.g., using a normal distribution) and, then, assesses it (evaluating the fitness of the current probability distribution) [31], [32]. Therefore, iteratively, a new population of individuals is generated sampling the current probability distribution, governed by the previously calculated parameters, and then evaluated.

This process is depicted in Algorithm 1, where: **population** denotes the list of individuals (default value: 100); **bpopulation** denotes the best chosen individuals from the population; **select_ratio** is the ratio for choosing the *bpopulation* (default value: 0.5); **reduction()** reduces the population based on *select_ratio*; **estimate_params()** adjust the distribution parameters (with mean (μ) and standard deviation (σ)); and **re_sample()** generates the new population based on the new parameters found.

Algorithm 1 Estimation of Distribution Algorithm

$population \leftarrow initialize(\mu, \sigma)$
$evaluate(population)$
while $condition\_not\_met$
  $bpopulation \leftarrow reduction(population, select\_ratio)$
  $(\mu, \sigma) \leftarrow estimate\_params(bpopoulation)$
  $population \leftarrow re\_sample(\mu, \sigma)$
  $evaluate(population)$
end while

Case 2 (Evolutionary Algorithm): EA is a generic population-based metaheuristic optimization algorithm that uses mechanisms inspired by biological evolution, such as reproduction, mutation, recombination, and selection. Candidate solutions to the optimization problem are the individuals of a population, and the evaluation function determines the quality of each solution. Evolution of the population then takes place after the repeated application of the above operators.

This process is depicted in Algorithm 2 [33], [34], where: **population** denotes a list of individuals (default value: 100); **cxpb** is the probability of crossing two individuals (default value: 0.5); **mutpb** is the mutation probability (default value: 0.3); **ngen** is the number of generations (default value: 12); **bNGF()** (buildNextGenerationFrom) computes the next generation population applying crossing and mutation operations to the selected population, validates the new individuals, and computes the statistics of this new population.

Algorithm 2 Evolutionary Algorithm

$evaluate(population)$
for g *in range* $(ngen)$
  $population = select(population, len(population))$
  $offspring = bNGF(population, cxpb, mutpb)$
  $evaluate(offspring)$
  $population = offspring$
end for

In our experiments we used a GA, which is an specific instance of an EA. In Table 7 the fundamental entities of a GA for the asymmetric indoor localization problem are defined.

TABLE 7. Fundamental entities of the genetic algorithm evaluated.

Entity	Definition
Individuals	The representative unit of an entity that will evolve to the optimum. In this case, individuals will be the different combinations of TxPower levels.
Gene	The minimum representation of a characteristic of an individual. In this case, each beacon will have six different TxPower levels.
Population	The set of individuals to be evaluated.
Crossing	This process consists of transmitting the genes from one generation to the next. In this case, new individuals are generated, inheriting some genes from their parents.
Mutation	This process involves the variation of genes in the genetic chain of the individual. The problem here is the change of a certain gene randomly in some individual(s) in the population.
Evaluation	This process consists of calculating or estimating how optimal (fitness) each individual of the population is. The accuracy of the algorithm with the individual parameters is obtained.
Selection	This process, as its name suggests, consists of separating the top optimal individuals for the next generation. It corresponds to the selection of individuals of the new generation that offer better accuracy.

With these initial definitions of both algorithms, the results obtained for both cases are discussed below.

B. METAHEURISTIC OPTIMIZATION RESULTS

As we already said above, the main challenge we faced with is to find an algorithm able to search for the optimal TxPower configuration and performing the minimum number of evaluations. In other words, the fewer evaluations a metaheuristic

TABLE 8. Experimental results using EDA and GA metaheuristic optimization algorithms. in bold are highlighted the results that obtained identical results as the brute-force search (see Table 6).

M-H	Model	TxPower	Acc.	ME	Eval.	Gen.
EDA	LoR	[6–1–2–3–5]	70.00	0.721	768	4
	LDA	[5–1–3–3–1]	73.60	0.611	782	5
	GNB	[6–1–3–3–1]	84.70	0.358	749	11
	MLP	[4–1–2–1–1]	66.90	0.892	785	12
	SVM	[6–6–3–3–5]	83.00	0.512	808	9
	DT	[6–1–3–3–3]	84.31	0.388	775	4
	k-NN	[6–1–2–3–1]	86.63	0.306	672	4
	RF	[6–1–3–3–3]	87.83	0.229	710	4
	ET	[4–1–3–3–5]	86.86	0.260	702	2
	GBM	**[6–1–3–3–5]**	**91.45**	**0.185**	**926**	**9**
	AB	[6–1–3–3–3]	87.68	0.254	808	5
	VC	[6–1–2–3–5]	88.62	0.263	866	6
GA	**LoR**	**[4–1–2–3–1]**	**70.85**	**0.647**	**334**	**6**
	LDA	**[6–1–3–3–5]**	**77.18**	**0.486**	**460**	**6**
	GNB	**[4–1–2–6–1]**	**86.01**	**0.289**	**439**	**7**
	MLP	**[4–1–4–6–1]**	**68.18**	**0.692**	**386**	**8**
	SVM	[6–6–3–3–5]	82.96	0.512	402	2
	DT	[6–1–3–3–5]	85.25	0.282	421	9
	k-NN	**[6–1–3–6–1]**	**87.78**	**0.264**	**368**	**2**
	RF	**[6–1–3–3–5]**	**89.84**	**0.228**	**408**	**6**
	ET	**[6–1–3–3–5]**	**89.25**	**0.197**	**341**	**4**
	GBM	**[6–1–3–3–5]**	**91.45**	**0.185**	**425**	**9**
	AB	[6–1–3–3–1]	87.88	0.291	358	5
	VC	**[6–1–3–3–5]**	**90.43**	**0.200**	**426**	**5**

algorithm performs the better, as long as the same solution to the problem is obtained. In this sense, Table 8 shows the results obtained with EDA and GA for each classification model. It is important to note that the value of the evaluation function used in both algorithms is the accuracy obtained for each specific classification model of each individual of the population, i.e., for each candidate solution to the problem. According to these results, we can observe how EDA needs more number of evaluations than GA, see column Eval, but both algorithms mainly need the same number of generations to converge, see column Gen. Also, in general EA obtains better TxPower configurations that EDA, obtaining in most of the cases (except for SVM, DT, and AB) the optimal solution. This fact can be observed when comparing Table 8 with 6.

C. HYPERPARAMETER TUNING

Tuning is the optimization or adjustment process for the hyperparameters of a model. It involves the comparison of cross-validation results for the selected metric under different types of adjustments. The objective is to choose the best combination of hyperparameters to maximize the chosen metric and accuracy for the best TxPower obtained by the metaheuristic algorithms. In our experiments, the "Grid-searchCV" function of the Scikit-learn library [30] has been used.

Table 9 lists all hyperparameters for each of the classification models assessed. Also, the search range of each hyperparameter and the selected value for the best accuracy results are shown.

Table 10 summarizes the results obtained with the GA algorithm with the selected optimized/tuned hyperparameters.

TABLE 9. Search range and selected hyperparameter for each model.

Model	Hyperparameters	Range	Selected
LoR	multi_class	['ovr', 'multinominal']	['multinomial']
	solver	['newton-cg', 'lbfgs', 'sag']	['newton-cg']
	C	[0.8, 0.9, 1.0, 1.1, 1.2]	[0.9]
	warm_start	[True, False]	[True]
LDA	solver	['svd', 'lsqr', 'eigen']	['eigen']
GNB	priors	[None]	[None]
MLP	hidden_layer_sizes	[10, 20, 50, 100]	[100]
	activation	['identity', 'logistic', 'tanh', 'relu']	['tanh']
	solver	['lbfgs', 'sgd', 'adam']	['adam']
	learning_rate	['constant', 'invscaling', 'adaptive']	['invscaling']
	alpha	[$10e^{-5}$, $10e^{-3}$, $10e^{-1}$, $10e^{1}$, $10e^{3}$]	[$10e^{-5}$]
SVM	shrinking	[True, False]	[True]
	kernel	['linear', 'poly', 'sigmoid', 'rbf']	['rbf']
	degree	[-, 3, 4, 5]	[-]
	function_shape	['ovo', 'ovr']	['ovo']
	C	[1, 2, 5, 10, 50, 100]	10
	gamma	['auto', 0.1, 0.01, 0.001]	[0.01]
k-NN	n_neighbors	[1, 3, 5, 7, 9, 11]	[5]
	weights	['uniform', 'distance']	['distance']
	algorithm	['ball_tree', 'kd_tree', 'brute']	['ball_tree']
DT	max_features	['sqrt', 'log2', None]	[None]
	splitter	['best', 'random']	['best']
	criterion	['gini', 'entropy']	['entropy']
	class_weight	[None, 'balanced']	[None]
	max_depth	[2, 3, 10, 50, 100]	[10]
RF	n_estimators	[5, 10, 15, 30]	[30]
	criterion	['gini', 'entropy']	['gini']
	max_features	[None, 'sqrt', 'log2']	['sqrt']
	class_weight	['balanced_subsample', None, 'balanced']	[None]
	max_depth	[2, 5, 10, 20]	[20]
	warm_start	[True, False]	[True]
ET	n_estimators	[10, 12, 15, 18, 20]	[20]
	criterion	['gini', 'entropy']	['gini']
	min_samples_leaf	[1, 2, 3, 4, 5]	[3]
	max_leaf_nodes	[3, 5, 7, 9, None]	[None]
	max_depth	[2, 3, 4, 5, None]	[None]
	max_features	[None, 'sqrt', 'log2']	['sqrt']
	class_weight	['balanced_subsample', None, 'balanced']	[None]
GBM	n_estimators	[100, 200, 250]	[100]
	max_depth	[3, 6, 9]	[3]
	learning_rate	[0.001, 0.1, 0.2, 0.3]	[0.1]
	max_features	['sqrt','log2', None]	['sqrt']
AB	algorithm	['SAMME', 'SAMME.R']	['SAMME']
	n_estimators	[50, 100, 500]	[500]
	learning_rate	[0.01, 0.1, 1.0, 2.0]	[2.0]
VC	voting	['hard', 'soft']	['soft']

Comparing evaluation metric values with those presented in Table 8, we can see how most models have improved in terms of accuracy and how most models have reduced the mean error. Specifically, for the VC classification model an improvement of 1.65% in terms of accuracy and a reduction of 0.019m in the mean error was obtained, outperforming the best result achieved so far by the GBM classification model in accuracy but not in mean error.

In addition, Figure 7 graphically represents these boxplot results. The variance for most of these latter results is considerably lower than the one reported in Figure 6(a).

TABLE 10. **Experimental results of the GA algorithm using tuned hyperparameters, for each classification model. best result is shown in bold.**

Model	TxPower Setting	Acc	ME	RT
LoR	[4–1–2–3–1]	76.71	0.487	10.001
LDA	[6–1–3–3–5]	77.60	0.481	0.031
GNB	[4–1–2–6–1]	86.01	0.289	0.020
MLP	[4–1–4–6–1]	72.21	0.613	3.660
SVM	[6–6–3–3–5]	83.00	0.512	2.881
DT	[6–1–3–3–5]	85.80	0.326	0.020
k-NN	[6–1–3–6–1]	87.87	0.249	0.031
RF	[6–1–3–3–5]	90.56	0.201	0.082
ET	[6–1–3–3–5]	91.03	0.209	0.050
GBM	[6–1–3–3–5]	91.87	**0.180**	3.269
AB	[6–1–3–3–1]	89.24	0.242	3.907
VC	[6–1–3–3–5]	**92.08**	0.181	3.365

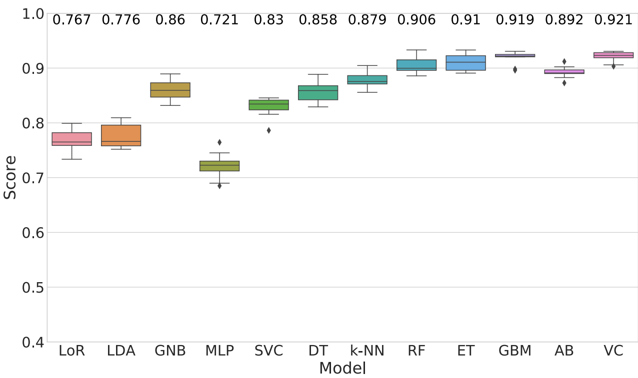


FIGURE 7. **Accuracy results of optimal asymmetric transmission power for the GA.**

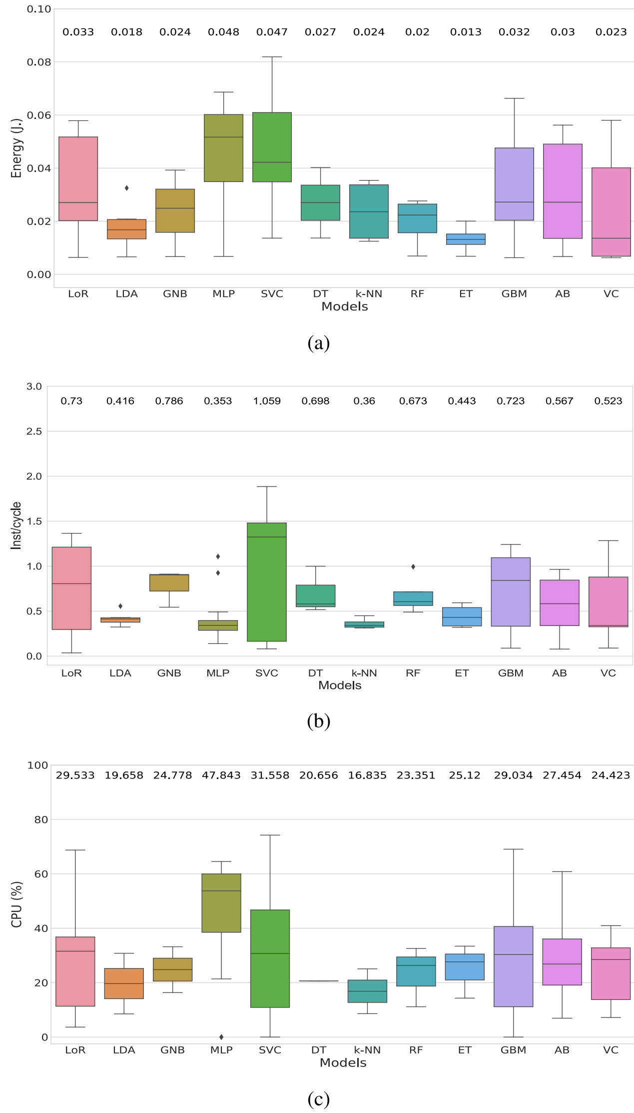


FIGURE 8. **Benchmark of the different classification models: (a) Energy consumption, (b) Inst/Cycle and (c) CPU.**

Furthermore, Table 10 clearly shows that the run time required, RT in seconds (s), to find the best combination of TxPower by GA is substantially shorter than the required by EDA and, obviously, by the brute-force search process. For instance, in the case of the GBM model, the GA performed 425 combinations in 3.269s for a total time of 1, 389.325s (23.155min). However, the EDA and brute-force, their RTs were 3, 027.094s (50.452min) and 25, 419.744s (423.662min), respectively.

Finally, we can observe how the boosted aggregation (bagging) models, i.e., RF and ET, obtained very good performance and a very low RT. The DT and *k*-NN models, although presenting a lower performance between 5% and 3% very closely by DT, *k*-NN, and RF. These four models are good alternatives for real-time localization, although.

D. BENCHMARKING RESULTS

Being able to evaluate the computational cost with respect to the accuracy of each model should provides us further insights on the computational requirements of the 12 models under study. Towards this end, we performed a benchmark test using the Perf software package on a computer equipped with 8GB RAM and an Intel i7 3.60GHz x8 processor.

Our computational cost analysis included the evaluation of the energy consumption in Joules (J), instructions per cycle (inst/cycle) and CPU percentage of usage (% CPU). Figure 8 shows the results of the benchmark test for the results obtained in Table 10.

Regarding energy consumption (Figure 8(a)), the ensemble algorithms, except the bagging algorithm, have a very high consumption for LoR, MLP, and SVM classification models. In Inst/cycle and % CPU, Figures 8(b) and 8(c), similar results were found. The best results wre reported for the ET model, followed very closely by DT, *k*-NN, and RF. These four models are good alternatives for real-time localization.

VII. CONCLUSIONS AND FUTURE WORK

This paper contributed to the mitigation of the MPF effect based on asymmetric TxPower setups. Twelve different SLA algorithms belonging to three different taxonomies have been studied. The results showed a remarkable improvement in the accuracy of the classification models. Therefore, to perform an optimized search between the different beacons and TxPower levels, two algorithms based on metaheuristics were evaluated: Estimation of Distribution Algorithm (EDA) and Genetic Algorithm (GA). The experimental results carried out exhibited that GA achieved optimal TxPower setup in most of the cases, reducing the number of evaluations considerably in comparison with the brute-force or exhaustive search approach.

To further improve the performance of the localization mechanism, a benchmark test was performed on energy and CPU consumption as well as Inst/cycle, the results of which confirmed that the boosted aggregation algorithm has a useful relationship between metrics and computational resource consumption. All studied algorithms, especially the non-linear and ensemble models, are a good option, although their very high computational resource consumption opens a new challenge in the actual deployment based on embedded devices.

Finally, the development of the distributed platform has the main purpose of developing indoor localization applications based on fog computing architecture. Traditional approaches use a cloud-like architecture for data storage and processing, while a future work of this research will perform data processing at the edge level, i.e., taking final decisions near to the receiver not in the cloud. Moreover, to minimize the energy consumption at the edge level, microcomputers can be used to make real-time decisions.

REFERENCES

[1] X. Zhao, Z. Xiao, A. Markham, N. Trigoni, and Y. Ren, "Does BTLE measure up against WiFi? A comparison of indoor location performance," in *Proc. Eur. Wireless 20th Eur. Wireless Conf.* Barcelona, Spain: VDE, 2014, pp. 1–6.

[2] Y. Yuan, C. Melching, Y. Yuan, and D. Hogrefe, "Multi-device fusion for enhanced contextual awareness of localization in indoor environments," *IEEE Access*, vol. 6, pp. 7422–7431, 2018.

[3] M. Sugano, T. Kawazoe, Y. Ohta, and M. Murata, "Indoor localization system using RSSI measurement of wireless sensor network based on ZigBee standard," *Wireless Opt. Commun.*, vol. 538, pp. 1–6, 2006.

[4] L. Pei, M. Zhang, D. Zou, R. Chen, and Y. Chen, "A survey of crowd sensing opportunistic signals for indoor localization," *Mobile Inf. Syst.*, vol. 2016, Apr. 2016, Art. no. 4041291.

[5] X. Wang, L. Gao, S. Mao, and S. Pandey, "CSI-based fingerprinting for indoor localization: A deep learning approach," *IEEE Trans. Veh. Technol.*, vol. 66, no. 1, pp. 763–776, Jan. 2017.

[6] A. S. Paul and E. A. Wan, "RSSI-based indoor localization and tracking using sigma-point Kalman smoothers," *IEEE J. Sel. Topics Signal Process.*, vol. 3, no. 5, pp. 860–873, Oct. 2009.

[7] J. Yim, C. Park, J. Joo, and S. Jeong, "Extended Kalman Filter for wireless LAN based indoor positioning," *Decis. Support Syst.*, vol. 45, no. 4, pp. 960–971, 2008.

[8] D. Magistro *et al.*, "A novel algorithm for determining the contextual characteristics of movement behaviors by combining accelerometer features and wireless beacons: Development and implementation," *JMIR Mhealth Uhealth*, vol. 6, no. 4, p. e100, 2018.

[9] R. Faragher and R. Harle, "Location fingerprinting with Bluetooth low energy beacons," *IEEE J. Sel. Areas Commun.*, vol. 33, no. 11, pp. 2418–2428, Nov. 2015.

[10] S. Ishida, Y. Takashima, S. Tagashira, and A. Fukuda, *Design and Initial Evaluation of Bluetooth Low Energy Separate Channel Fingerprinting*. Cham, Switzerland: Springer, 2018, pp. 19–33. doi: 10.1007/978-3-319-70636-8_2.

[11] R. Chavez-Romero, A. Cardenas, M. Maya, and A. Sanchez, and D. Piovesan, "Camera space particle filter for the robust and precise indoor localization of a wheelchair," *J. Sensors*, vol. 2016, Mar. 2016, Art. no. 8729895.

[12] M. Castillo-Cara, J. Lovón-Melgarejo, G. Bravo-Rocca, L. Orozco-Barbosa, and I. García-Varea, "An analysis of multiple criteria and setups for Bluetooth smartphone-based indoor localization mechanism," *J. Sensors*, vol. 2017, Oct. 2017, Art. no. 1928578.

[13] A. Rahim, R. Dimitrova, and A. Finger, *Techniques for Bluetooth Performance Improvement*. 2017.

[14] M. Brunato and R. Battiti, "Statistical learning theory for location fingerprinting in wireless LANs," *Comput. Netw.*, vol. 47, no. 6, pp. 825–845, 2005.

[15] M. Castillo-Cara, J. Lovún-Melgarejo, G. Bravo-Rocca, L. Orozco-Barbosa, and I. García-Varea, "An empirical study of the transmission power setting for bluetooth-based indoor localization mechanisms," *Sensors*, vol. 17, nos. 6, p. 1318, 2017.

[16] W. Kang and Y. Han, "SmartPDR: Smartphone-based pedestrian dead reckoning for indoor localization," *IEEE Sensors J.*, vol. 15, no. 5, pp. 2906–2916, May 2015.

[17] V. C. Paterna, A. C. Augé, J. P. Aspas, and M. A. P. Bullones, "A bluetooth low energy indoor positioning system with channel diversity, weighted trilateration and Kalman filtering," *Sensors*, vol. 17, no. 12, p. 2927, 2017.

[18] J. Powar, C. Gao, and R. Harle, "Assessing the impact of multi-channel BLE beacons on fingerprint-based positioning," in *Proc. Int. Conf. Indoor Positioning Indoor Navigat. (IPIN)*, Sep. 2017, pp. 1–8.

[19] S. Shukri *et al.*, "Analysis of RSSI-based DFL for human detection in indoor environment using IRIS mote," in *Proc. 3rd Int. Conf. Electron. Design (ICED)*, Aug. 2016, pp. 216–221.

[20] E. Leitinger, P. Meissner, C. Rüdisser, G. Dumphar, and K. Witrisal, "Evaluation of position-related information in multipath components for indoor positioning," *IEEE J. Sel. Areas Commun.*, vol. 33, no. 11, pp. 2313–2328, Nov. 2015.

[21] T.-T. Lu, S.-C. Yeh, and C.-Y. Chen, "A study of indoor positioning systems using iBeacons with different transmission power levels," *J. Chin. Inst. Eng.*, vol. 40, no. 6, pp. 525–535, 2017.

[22] X.-Y. Lin, T.-W. Ho, C.-C. Fang, Z.-S. Yen, B.-J. Yang, and F. Lai, "A mobile indoor positioning system based on ibeacon technology," in *Proc. 37th Annu. Int. Conf. IEEE Eng. Med. Biol. Soc. (EMBC)*, Aug. 2015, pp. 4970–4973.

[23] P. Kriz, F. Maly, and T. Kozel, "Improving indoor localization using bluetooth low energy beacons," *Mobile Inf. Syst.*, vol. 2016, Mar. 2016, Art. no. 2083094.

[24] X. Wang, L. Gao, S. Mao, and S. Pandey, "CSI-based fingerprinting for indoor localization: A deep learning approach," *IEEE Trans. Veh. Technol.*, vol. 66, no. 1, pp. 763–776, Jan. 2017.

[25] G. G. Anagnostopoulos, M. Deriaz, and D. Konstantas, "Online self-calibration of the propagation model for indoor positioning ranging methods," in *Proc. Int. Conf. Indoor Positioning Indoor Navigat. (IPIN)*, Oct. 2016, pp. 1–6.

[26] L. Chen, B. Li, K. Zhao, C. Rizos, and Z. Zheng, "An improved algorithm to generate a Wi-Fi fingerprint database for indoor positioning," *Sensors*, vol. 13, no. 8, pp. 11085–11096, 2013.

[27] R. Faragher and R. Harle, "An analysis of the accuracy of bluetooth low energy for indoor positioning applications," in *Proc. 27th Int. Tech. Meeting Satell. Division Inst. Navigat. (ION GNSS+)*, vol. 812, Tampa, FL, USA, 2014, pp. 201–210.

[28] J. Inc. (2018). *Beacon IB0004-N Plus*. Accessed: Dec. 30, 2018. [Online]. Available: https://www.jaalee.com/

[29] Trendnet. (2018). *Micro Bluetooth USB Adapter*. Accessed: Dec. 30, 2018. [Online]. Available: https://www.trendnet.com/products/USB-adapters/TBW-107UB/

[30] F. Pedregosa *et al.*, "Scikit-learn: Machine learning in python," *J. Mach. Learn. Res.*, vol. 12, pp. 2825–2830, Oct. 2011.

[31] E. Bengoetxea, "Estimation of distribution algorithms, *Ecole Nat. Supérieure des Télécommun. Paris*, vol. 2, no. 4, pp. 43–76, 2002.

[32] M. Pelikan, M. W. Hauschild, and F. G. Lobo, "Estimation of distribution algorithms," in *Springer Handbook of Computational Intelligence*, J. Kacprzyk and W. Pedrycz, Eds. Berlin, Germany: Springer, 2015, pp. 899–928. doi: 10.1007/978-3-662-43505-2_45.

[33] T. Bäck, D. B. Fogel, and Z. Michalewicz, *Evolutionary Computation 1: Basic Algorithms and Operators*, vol. 1. Boca Raton, FL, USA: CRC Press, 2000.

[34] F.-A. Fortin, F.-M. De Rainville, M.-A. Gardner, M. Parizeau, and C. Gagné, "DEAP: Evolutionary algorithms made easy," *J. Mach. Learn. Res.*, vol. 13, pp. 2171–2175, Jul. 2012.

JESÚS LOVÓN-MELGAREJO received the B.S. degree in computer science from the Universidad Nacional de Ingeniería (UNI), Perú, in 2016, and the M.Sc. degree from the Université de Toulouse, France. He is currently a member of the Intelligent Ubiquitous Technologies Lab (IUT-SCi), UNI. His research interests include artificial intelligence, parallel computing, and analysis of algorithms.

MANUEL CASTILLO-CARA received the Ph.D. degree from the Universidad de Castilla-La Mancha, in 2018. He has been working on university educational issues at the Computer Science as an Associate Professor and the Head of Intelligent Ubiquitous Technologies (IUT-SCi), Universidad Nacional de Ingeniería. His current research interests include intelligent ubiquitous technologies, especially on in wireless sensor networks, distributed computing, pattern recognition, and artificial intelligence.

OSCAR HUARCAYA-CANAL received the B.Sc. degree in computer science from the Universidad Nacionald de Ingeniería (UNI), Lima, Perú, in 2018. He is a member of the Intelligent Ubiquitous Technologies Lab, UNI. He is researching optimization metaheuristics applied to data science. His research interests include machine learning and evolutionary computation.

LUIS OROZCO-BARBOSA (M'90) received the Doctorat de l'Université degree from the Université e Pierre et Marie Curie, France, in 1987. From 1987 to 2001, he was a Faculty Member with Electrical and Computer Engineering Department, University of Ottawa, Ottawa, ON, Canada. In 2002, he joined the Department of Computer Engineering, Universidad de Castilla-La Mancha, Spain. His current research interests include Internet protocols, wireless sensor communications, and the IoT technologies.

ISMAEL GARCÍA-VAREA received the B.Sc. degree in computer science from the Universidad de Castilla-La Mancha (UCLM), in 1991, and the M.Sc. degree in computer science and the Ph.D. degree in pattern recognition and artificial intelligence from the Universitat Politècnica València, Spain, in 1996 and 2003, respectively. In 1999, he joined the Computing Systems Department, UCLM, where he is currently an Associate Professor. At the same time, he joined the Research Group Intelligent Systems and Data Mining (SIMD), leading the research lines of natural language processing and autonomous robotics.