

Research Article

Optimal Modeling of Wireless LANs: A Decision-Making Multiobjective Approach

Tomás de Jesús Mateo Sanguino  and Jhon Carlos Mendoza Betancourt

Departamento de Ingeniería Electrónica, Sistemas Informáticos y Automática, University of Huelva, Ctra. Huelva-La Rábida S/N, Palos de la Frontera, 21819 Huelva, Spain

Correspondence should be addressed to Tomás de Jesús Mateo Sanguino; tomas.mateo@diesia.uhu.es

Received 24 November 2017; Accepted 19 February 2018; Published 26 March 2018

Academic Editor: Eloy Irigoyen

Copyright © 2018 Tomás de Jesús Mateo Sanguino and Jhon Carlos Mendoza Betancourt. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Communication infrastructure planning is a critical design task that typically requires handling complex concepts on networking aimed at optimizing performance and resources, thus demanding high analytical and problem-solving skills to engineers. To reduce this gap, this paper describes an optimization algorithm—based on evolutionary strategy—created as an aid for decision-making prior to the real deployment of wireless LANs. The developed algorithm allows automating the design process, traditionally handmade by network technicians, in order to save time and cost by improving the WLAN arrangement. To this end, we implemented a multiobjective genetic algorithm (MOGA) with the purpose of meeting two simultaneous design objectives, namely, to minimize the number of APs while maximizing the coverage signal over a whole planning area. Such approach provides efficient and scalable solutions closer to the best network design, so that we integrated the developed algorithm into an engineering tool with the goal of modelling the behavior of WLANs in ICT infrastructures. Called WiFiSim, it allows the investigation of various complex issues concerning the design of IEEE 802.11-based WLANs, thereby facilitating design of the study and design and optimal deployment of wireless LANs through complete modelling software. As a result, we comparatively evaluated three target applications considering small, medium, and large scenarios with a previous approach developed, a monoobjective genetic algorithm.

1. Introduction

Implementing Wi-Fi networks with the best use of resources while offering the best service to users requires a careful planning. WLANs can range from relatively simple installations to very complex and intricate designs so a well-documented plan must be outlined before a wireless infrastructure can be deployed [1]. In practice, network technicians must survey the space towards deciding on the WLAN arrangement, whose process involves gauging the location of APs, clients, and obstacles. This is traditionally reduced to achieve the maximum Wi-Fi coverage as decision criterion, which is translated into varying the AP placements based on signal strength measurements. Nevertheless, it is not merely drawing coverage circles on a plane and the difficulty increases when requiring a realistic view of the whole problem [2]. This indeed includes several factors such as the area morphology

(i.e., verticality and horizontality of planes), number of clients populating the WLAN (i.e., distribution, density), type of IEEE 802.11 technology (i.e., modulation scheme and frame management), effective isotropic radiated power (EIRP), and physical distance and obstacle materials (i.e., water, glass, plastic, metal, wood, and concrete). The traditional deployment is consequently no effective for practitioners due to requirements in time and cost, which may drastically reduce the WLAN performance and usability due to operability problems concerning design (e.g., interferences and frame collisions due to home devices, hidden node problem) [3].

To help reduce this gap, this paper presents WiFiSim, an engineering modelling software being developed as part of an ongoing research project for the study and design of WLANs [4–6]. The purpose of this software is to model the behavior and performance of communication networks based on the IEEE 802.11 standard. The interest lies in the realism of the

WiFiSim simulations, which provide a high level of interactivity and visual information with easy-to-interpret results through a configurable and intuitive GUI. In this paper we propose a further development beyond the study and design of WLANs: the problem of optimally deploying APs to cover Wi-Fi clients within a complex space. So this manuscript intended three significant contributions: (1) to present a new intelligent modelling approach for WLAN deployment based on a multiobjective genetic algorithm; (2) to examine its applicability into small, medium, and large scenarios; and (3) to integrate the algorithm with the previous developments and provide a complete decision-making software for ICT technicians. To achieve this goal, we propose an evolutionary genetic strategy based on nondominated sorting genetic algorithm II (NSGA-II) where the optimization approach is applied into a demanding WLAN infrastructure.

To this end, the paper is organized as follows. Section 2 introduces the tools and approaches for WLAN deployment existing in literature. Section 3 is devoted to describing the evolutionary genetic strategy followed to develop the optimization algorithm. Section 4 presents the experimentation conducted in three representative case studies. Finally, the paper provides the results and future works.

2. State of the ART

Typically, the location of APs in Wi-Fi networks is manually estimated from their power transmission. From this parameter, APs are placed at regular intervals throughout the space with a predefined distance between adjacent APs. However, this approach idealizes the signal coverage that, in reality, often finds complex environments with a wide variety of obstacles and materials. This implies designs that may result in a deficient configuration of APs due to poor or excessive coverages. When applied to large scenarios, this approach may bring a significant cost overrun. To address these shortcomings, site surveys must be combined with software modelling tools that help improve and simplify the human process [7]. However, these tools need to have a high degree of realism and simulation capabilities to be really helpful (e.g., modeling of physical spaces and network behavior in Layer 1 and Layer 2 of the OSI model). For this reason, choosing a modelling software to study and design optimal WLANs can become hard due to the large number of existing tools [8].

2.1. Modelling Software for WLAN Deployment. At present, there is a large number of applications for designing and/or planning Wi-Fi networks. Examples of research in scientific literature using these applications include NetStumbler®, a survey-site tool that facilitates the detection of WLANs using the IEEE 802.11 protocol [9]; Wi-Fi Analytics Tool™, a software that provides advanced signal strength graphs and analyzes Wi-Fi channels to optimize the Wi-Fi network setup [10]; Wi-Spy DBx, a RF spectrum analyzer designed for troubleshooting Wi-Fi issues with nearby interfering devices in the 2.4 and 5 GHz bands [11]; Ekahau HeatMapper™, an auditory tool and Wi-Fi site-survey software for home use [12]; NetSpot, a site-survey and analyzing tool that helps improve the Wi-Fi signal strength and to boost the network

speed [13]; Acrylic® WiFi Heatmaps, a site-survey and audit software aimed at generating awesome Wi-Fi heatmaps and editable reports on the RF spectrum coverage [14]; Wolf WiFi Pro®, a Wi-Fi device management software and pre-deployment toolkit for wireless professionals able to detect failure scenarios [15]; Bat-Planner, a basic planning suite for IEEE 802.11-based wireless networks [16]; TamoGraph®, a site-survey tool for collecting, visualizing, analyzing, and reporting Wi-Fi data [17]; VisualRF Plan, a wireless management suite that helps to model the RF environment and the underlying wired topology in a visual way [18]; RF3D Wifiplanner2, a modelling tool for planning and upgrading WLANs based on the study of RF signals [19]; AirMagnet Survey®, an accurate and flexible solution for planning indoor and outdoor WLANs [20]; WiTuners™, a Wi-Fi tool for site survey, automated deployment, and auditing [21]; and Ekahau Site Survey™, a professional software toolkit for site survey, spectrum analysis, and Wi-Fi planning [22], to name a few tools.

In general, all the above tools exemplify design and inspection software for wardriving, site survey, data collecting, and planning. Among the drawbacks, these applications (i) are mostly commercial software with a pay-per-use license agreement, (ii) are often natively available only for Windows platforms, (iii) do not always include structures or materials for obstacle modelling, (iv) only allow multifloor design through 2D maps, (v) do not integrate habitually algorithms for optimal WLAN deployment, and (vi) support their designs in a two-step process (i.e., definition of site requirements and planning based on RF propagation). Among them, AirMagnet Survey, WiTuners, and Ekahau Site Survey are remarkable exceptions. On the one hand, they automatically plan the AP positions and quantity needed to ensure a minimum coverage. On the other hand, they validate their designs with data collecting by auditing the real environment.

As a main disadvantage, these tools use proprietary algorithms for WLAN deployment based on RF planning (i.e., patent protected). This implies charging a licensing fee that can be fairly expensive (especially in comparison to open-source software), to make the owner too dependent on the developer (possibly less adaptable to the constantly changing needs of users) and to make the mechanics of the algorithms opaque to viewing (i.e., there is no testable information) [23]. The importance of utilizing evolutionary approaches, instead of providing a single solution as in the case of the RF-based algorithms, is to procure a set of optimal solutions to facilitate broader decision-making for users. Moreover, the above tools lack a modelling block to assist engineers in the study of wireless traffic in Layer 2. Both capabilities combined are critical when troubleshooting wireless communication issues to deploy efficient and useful WLANs. This way, the design process is improved by a cycle of three steps, much more creative, based on the generation of alternative solutions and analysis and selection of the one considered more fitting.

To introduce the developed modelling software, the main features and properties of WiFiSim are compared with the above-mentioned tools in Table 1. In sum, the pros of WiFiSim are (i) to facilitate for engineers the WLAN deployment by providing a set of optimal solutions closer to the best

TABLE 1: Comparison of features and capabilities of various applications for the study and deployment of wireless networks.

Software name	IEEE 802.11 standard	Study of Layer 1	Study of Layer 2	Obstacle modelling	Type of coverage	Main application	Automatic planning	Native platform	Size (MB)	Cost (€)
NetStumbler	a/b/g	Signal strength	No	No	2D view	Wardriving	No	Windows	1.30	Free
Wi-Fi Analytics Tool	a/b/g/n/ac	Signal strength	No	No	2D view	Wardriving	No	Windows & Android	1.8 & 0.5	Free
Wi-Spy DBx	a/b/g/n	RF spectrum	No	No	2D view	Wardriving* & RF audit	No	Windows	7.60	450–1000
Ekahau HeatMapper	a/b/g/n	Signal strength	No	No	2D map	Site survey	No	Windows	100.0	Shareware [†]
NetSpot	a/b/g/n/ac	Signal strength	No	No	2D map	Site survey	No	Windows & Mac OS X	17.0	50–500
Acrylic WiFi Heatmaps	a/b/g/n/ac	Signal strength	No	No	2D map	Site survey	No	Windows	15.9	879–1119 [●]
Wolf WiFi Pro	a/b/g/n	Signal strength	No	No	2D map	Site survey & AP management	No	Android	0.72	0–50
Bat-Planner	a/b/g/n	Signal strength	No	✓	3D view	Site survey & planning	No	Windows	N/A	Free*
TamoGraph	a/b/g/n/ac	Signal strength	No	✓	2D map	Site survey, data collecting* & planning	No	Windows	81.0	750–1000
VisualRF Plan	N/A	Signal strength	No	✓	2D map	Predeployment	No	Windows	206.9	Free
RF3D Wifiplanner2	a/b/g/n	Signal strength	No	✓	2D map	Planning	No	Windows	15.9	Shareware [♦]
AirMagnet Survey	a/b/g/n/ac	RF data & signals	No	✓	2D map & 3D view	Site survey, data collecting* & planning	RF-based algorithm	Windows	477.0	1800
WiTuners	a/b/g/n/ac	Signal strength	No	✓	2D map	Site survey, data collecting* & planning	RF-based algorithm	Windows & Android	120.1 & 7.20	760–2400 & 0–500
Ekahau Site Survey	a/b/g/n/ac	Signal strength & RF spectrum	No	✓	2D map	Site survey, data collecting* & planning	RF-based algorithm	Windows & Mac OS X	338.0	3395
WiFiSim	a/b/g/n	Signal strength	CSMA/CA RT S/CTS	✓	2D & 3D maps	Modelling & planning	SSGA & NSGA-II	Multiplatform	3.18	Free

[†]Free version with limited features; [♦]15-day trial version; *special hardware required; [●]1-year license. N/A = not available.

network design and (ii) to study WLANs in both Layer 1 and Layer 2 to avoid several communication issues concerning the PHY and MAC layers (e.g., throughput, channel utilization, frame collision, frame delay, queue length and delay, medium access delay, and jitter and hidden node problem). As far as we know, these are the main differentiating capabilities regarding other tools, whose modelling process allows practitioners to improve complex network designs. As a result, WiFiSim has been utilized by professionals on ICT and students and teachers at the University of Huelva to improve the learning and teaching of computer networking degrees over the past seven years (i.e., 10 professionals, 110 students, and 4 teachers since the 2011/12 academic year).

2.2. Approaches for WLAN Optimization. Early approaches based on computer models consisted in exhaustive searches of AP positions under strong restrictions, thus preventing being useful in more general situations [24]. Traditional forms of infrastructure planning, as used for mobile cellular networks, produce acceptable results but are generally considered too costly for wireless networks [25]. In this field, optimization techniques range from Hill climbing, random walk (RW), simulated annealing (SA), and Tabu search (TS) to genetic algorithms (GA). Hill climbing, a mathematical optimization technique within local search methods, was also applied along to metaheuristic neighborhood search algorithms (e.g., SA or TS) to prevent being trapped in local optimums. However, mathematical programming techniques are usually less preferred than metaheuristic approaches due to the difficulty for adapting math expressions to generic case studies. Moreover, TS and GA proved to be techniques with better performance than other local search approaches such as RW or SA. However, TS suffer from scalability and evolutionary genetic strategies depend largely on tuning parameters for the application [26]. Both issues—scalability and parameter tuning—were effectively addressed through an agent-based optimization approach [27] and a combination of pruning and neighborhood search algorithms [28], respectively.

Initial investigations on GAs led to deploying a single AP through a simple implementation [29]. Subsequent works were focused on modeling WLAN scenarios by increasing the number of APs [30]. GAs then evolved to include new factors such as the effect of different obstacles on the coverage performance [31]. In general, monoobjective strategies centered on optimizing a target function are more popular because of flexibility, robustness, and ease of implementation but may not be realistic enough when several design goals are sought to meet (e.g., number of APs, overall SNR, or throughput). For this reason, most of the real-world optimization problems are multiobjective in nature, which often have two or more target functions that must be met simultaneously and possibly conflict with each other. This causes having a set of optimal solutions (i.e., Pareto solutions) instead of having a single outcome. On the other hand, monoobjective algorithms may require being run every time to obtain multiple solutions. These drawbacks were addressed in this field thanks to a multiobjective genetic algorithm (MOGA) approach. Such is the case of an improved adaptive

genetic algorithm (IAGA) for WLAN deployment based on attaining the AP quantity and signal route wastage as minimum as possible [32]. Analogously, a multiobjective strategy was proposed to obtain various optimal placement configurations for different numbers of APs based on the SNR region [33]. Although closer to the work being proposed, as main disadvantage, these algorithms were not provided as a modelling software for the complete study of the WLAN operation in Layer 1 and Layer 2 (i.e., not interactive and exhaustive).

In this paper, we specifically used NSGA-II, one of the most representative approaches of MOGAs due to its ability to find multiple Pareto solutions in a simple execution [34]. As a main advantage, this approach allows easily dealing with concave and discontinuous Pareto boundaries, combined with a crowding operator, allowing obtaining a wider set of optimal solutions than other GAs of first and second generation as the niched Pareto genetic algorithm (NPGA) [35], nondominated sorting genetic algorithm (NSGA) [36], strength Pareto evolutionary algorithm (SPEA) [37], Pareto archived evolution strategy (PAES) [38], Pareto envelope-based selection algorithm (PESA) [39], microgenetic algorithm (Micro-GA) [40], improved niched Pareto genetic algorithm (NPGA2) [41], improved envelope-based selection algorithm (PESA-II) [42], or improved strength Pareto evolutionary algorithm (SPEA2) [43], among others.

3. Modelling System

The first implementation of WiFiSim—acronym for Wireless Fidelity Simulator—consisted of an intelligent modelling software developed in Java™ with the Eclipse framework. WiFiSim allows modeling various parameters in the PHY and MAC layers of the OSI model with support for the IEEE 802.11a/b/g/n standards, including stand-alone configuration of APs and wireless clients (e.g., interbeacon frame, rate and sensitivity, Cartesian coordinates, transmission power, packet size, RTS threshold, and packet load distribution), support for a customized library of materials, and control of the medium access (i.e., CSMA/CA mechanism, RTS/CTS mechanism, and back-off algorithm). The validation of models in WiFiSim is done in a similar way as in AirMagnet Survey, WiTuners, and Ekahau Site Survey but using complementary tools, for instance, WiFi Analyzer. This application allows us to manually conduct real WLAN tests, such as measuring the signal sensibility, antenna aiming, or APs detection [44]. Thus, once nodes and obstacles are added to a wireless scenario, we can simulate network models with a high degree of realism and study signal loss-related problems. Further details are described in [4, 5].

In order to facilitate the graphical deployment of WLANs, WiFiSim was firstly extended with a steady-state genetic algorithm (SSGA). This allowed users to dynamically work on 3D maps and specify whether the APs must be automatically arranged anywhere inside the buildings or fixed on the walls according to common ICT installation requirements. The main advantage of using SSGA is that multiple optimal placement solutions can be obtained for the same configuration of APs (i.e., from 1 AP to 8 APs). This provides more

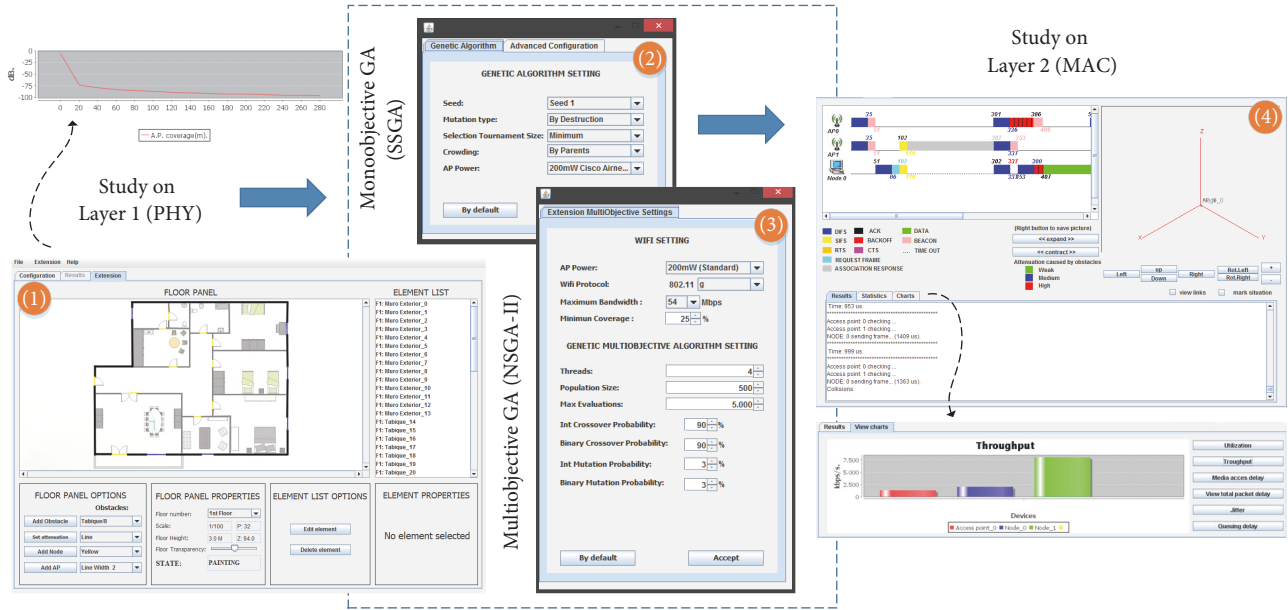


FIGURE 1: WiFiSim's control panel: (1) graphical deployment for Layer 1, (2) steady-state GA module, (3) multiobjective GA module, and (4) study of Layer 2.

alternatives to the network designer than automatic RF-based algorithms. Moreover, SSGA is able to automatically plan APs in both empty spaces (i.e., when the nodes are not known a priori) and populated environments (i.e., when the nodes are fixed beforehand). Hence, SSGA seeks to optimize the coverage when designing empty planes or maximize it on customer locations according to the user requirements. Briefly explained, SSGA consisted in the following steps: (1) begin with a selection of candidate solutions (i.e., positions for a specific number of APs), (2) evaluate the solutions according to the maximum coverage as objective, (3) evolve these solutions by using parent selection, crossover, mutation, and replacement with subsequent generation improvements as long as the best solution does not change, and (4) provide those best solutions to the user. Additional details can be found in [6].

3.1. Multiobjective Genetic Algorithm. As a complementary strategy to SSGA, a MOGA module has been integrated into WiFiSim. The major improvement of using MOGA compared to SSGA is that multiple optimal placement configurations consisting of different numbers of APs can be simultaneously obtained from a single run. That way, the functionalities of WiFiSim were extended by not only selecting up to 8 APs with SSGA but also collecting 32 simultaneous solutions per WLAN design with NSGA-II. It gives users a greater range of optimization and experimentation as shown in Figure 1.

The modelling technique was developed using jMetal, a Java framework for developing metaheuristic algorithms [45]. NSGA-II basically consists of a parent population P_t and a descendant population Q_t , both of size N , combined to form a new population $R_t = P_t \cup Q_t$ of size $2N$. While P_t is randomly created based on an initialization method, Q_t is generated from P_t using a tournament, crossing, and mutation process.

Then R_t is classified into Pareto fronts (F_i) according to a nondominated ordering process and a fitness function such that $R_t = \{F_1, F_2, F_3, \dots, F_i\}$. Instead of randomly choosing the Pareto fronts to form the next generation of size N , NSGA-II takes the best solution fronts and discards the worst fronts. This is efficiently done by computing the crowding distance (d_i) of their solutions, meaning the value of the search space around a solution that is not occupied by another solution in the population:

$$d_j^m = d_j^m + \frac{f_m^{(j_{+1})} - f_m^{(j_{-1})}}{f_m^{\max} - f_m^{\min}}, \quad (1)$$

where m stands for each of the objective functions and j means the solution taken from the Pareto front for which $j + 1$ and $j - 1$ stand for the neighboring solutions. As a result, NSGA-II ensures the best and more diversity results since the solutions are distanced from each other when the population converges towards the optimal Pareto front after several generational cycles. The problem formulation and the genetic operators used are discussed in the following sections.

3.2. Map Structure and Objectives. The floor maps were codified in NSGA-II by means of a cell division method. In the case of the SSGA, maps were divided—depending on the floor shape and obstacles available—into square cells, rectangular cells, and mini-cells with resolution up to $1 \times 1 \text{ m}^2$ (i.e., 32×32 pixels by default). The NSGA-II approach was improved, treating the cells closer to reality by using not only standard cells but also triangular and polymorphic shapes. This resulted in a more efficient cell division method since it completely covers the whole useful space (Figure 2).

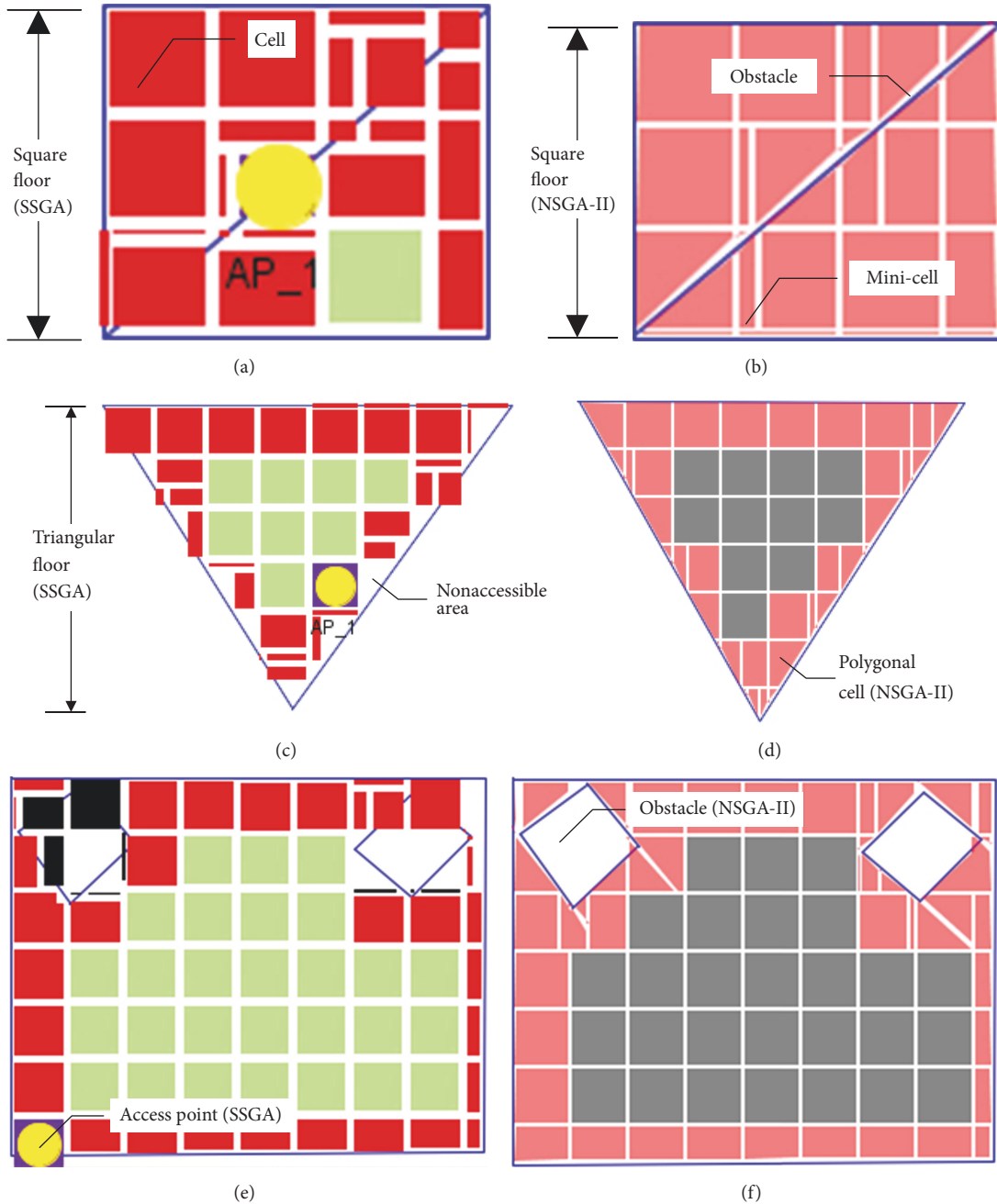


FIGURE 2: Comparison of the grid structure in SSGA (left) and NSGA-II (right) where red, green, and grey cells stand for covered areas: (a)-(b) square building with diagonal obstacle, (c)-(d) triangular building, and (e)-(f) rectangular building with inner obstacles (e.g., pillars).

There were two optimization objectives in this problem: minimize the number of APs and maximize the number of cells covering the whole planning area. These objectives were in conflict since reducing the number of APs inevitably implies decreasing the average signal as well, and vice versa. The first decision criterion was to minimize the number of APs to reduce the infrastructure cost as much as possible. As second decision criterion, NSGA-II computes the signal attenuation by distance and obstacle at each cell for every AP configuration (i.e., from 1 to 32 APs). The free space path

loss (FSPL) is calculated as a function of the distance and frequency, whose amount of signal loss is given in decibels by

$$\text{FSPL} = 20 \log_{10}(d) + 20 \log_{10}(f) + 32.44, \quad (2)$$

where d is the distance from the transmitter (in Km) and f is the frequency (in MHz). The loss signal by obstacle is then subtracted to (2) and the total power received at each cell is modeled as follows:

$$P_{rx} = P_{tx} - (\text{FSPL} - L_o), \quad (3)$$

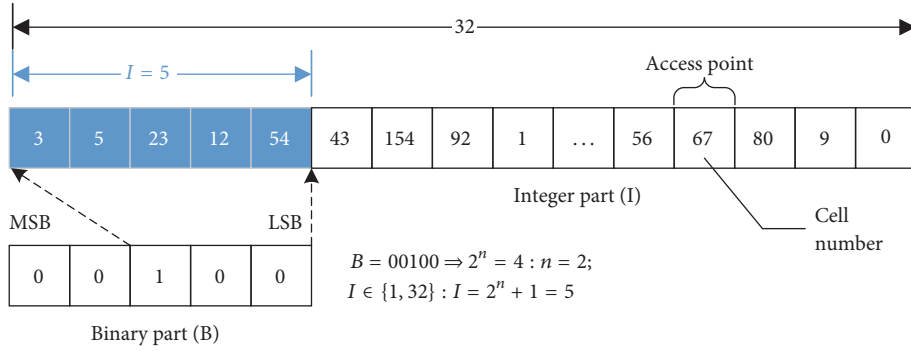


FIGURE 3: Coding scheme of the chromosome utilized in NSGA-II.

where P_{rx} is the power received at each cell by every AP, P_{tx} is the power emitted for each AP, and L_o is the loss signal due to the obstacle presence. Once the power signal is calculated at each map cell, two specific user constraints defined in the algorithm options are evaluated. These are the maximum bandwidth and Wi-Fi protocol. Since the rate and modulation scheme used in the WLAN set the media sensibility (i.e., rate in Mbps versus power in dBm), this serves to filter out the solution calculated by the algorithm in (3) and satisfy the design constraints. Therefore, the planning of the AP location plays a key role in maximizing the WLAN transmission coverage and communication throughput.

3.3. Coding Scheme and Chromosome Selection. The number of APs and their locations were concatenated to form a chromosome (Figure 3). We modified jMetal to use a double chromosome structure in NSGA-II consisting of a binary part and an integer part that codifies the two decision variables. The first one stands for 5 bits whose equivalent decimal value stands for the number of APs to be evaluated in the integer part of that chromosome. The second part contains 32 values that identify the cell number where the APs will be located. The number of cells will depend on the map size, number of floors, and obstacle shapes.

The chromosome selection is based on a fitness function and the stochastic universal sampling selection, where each individual of the population is valued in a nondominated Pareto rank after a selection process, becoming part of the next generation of those chromosomes with lower rank value [46]. In case of individuals with the same rank value, a crowding operator based on (1) is applied to discriminate individuals. The lower the value of the crowding operator, the greater the diversity of solutions that a chromosome has.

3.4. Mutation and Crossover Operators. The mutation process is applied independently to the binary and integer part of the chromosomes according to values of probability that initially are chosen very low (e.g., 3%). Specifically, we have used the integer flip mutation method for the integer part, which takes random positions of a chromosome to change their values. In case of generating a chromosome with more than one AP in the same cell position, the algorithm will consider them as the same AP and filter out this solution

in favor of others presenting the same number of APs but in different map cells. As a result, the algorithm discards those anomalous chromosomes before passing to the next generation. Regarding the binary part, we used a bit flip method with similar mutation to the integer flip technique but considering the range [0, 1].

The crossover process is also applied to the integer and binary parts severally. In particular, a uniform crossover scheme (UX) was applied to the integer part in which the values of two chromosomes are compared individually and swapped with a fixed probability, typically 0.5. Regarding the binary part, we used a one-point crossover operator in which all the values from a point of a chromosome are interchanged with the corresponding values in the other chromosome compared. Both crossover operators were entirely programmed in jMetal for this project.

3.5. Modelling Improvements. Beyond being an evolutionary genetic algorithm, NSGA-II was conceived to exploit its full capability as an intelligent modelling software. To this end, NSGA-II was integrated into WiFiSim to afford the parameter configuration of the multiobjective algorithm. First, because the generation, mutation, crossing, and selection of the population are processes cyclically done by NSGA-II, we end the algorithm execution by setting a maximum number of evaluations. This value, fixed to 5,000 by default as depicted in Figure 1, can be customized by users to experiment with different results. Once the algorithm finishes, it enters the process of ranking solutions, filters out those that are anomalous, and sorts them according to the number of APs that compose it. Hence, users can move in an orderly way from one solution to another—by means of a cursor—to study the arrangement of up to 32 access points. Conversely, in SSGA, when a replacement exists after an iteration, the algorithm reevaluates the population to determine whether it improves or not. In case of remaining stuck during 500 iterations, SSGA ends.

Secondly, NSGA-II allows search for optimal solutions based on requiring a minimum percentage of allowable coverage in the WLAN infrastructure according to a Wi-Fi bandwidth and technology (e.g., 1, 6, 11, 54, 108, 130, 150, or 300 Mbps for IEEE 802.11n). In this way, NSGA-II models the relationship between transmission rate (Mbps) and

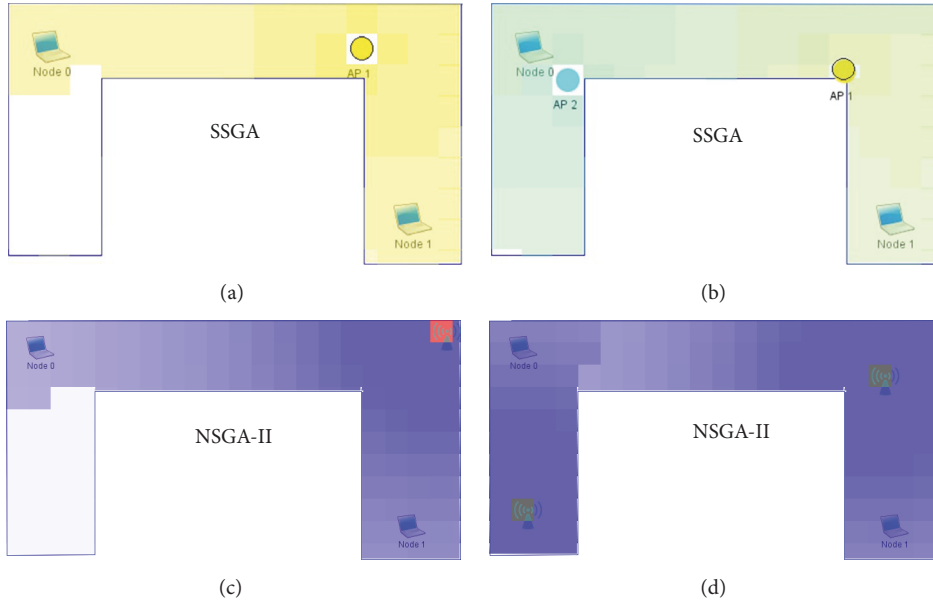


FIGURE 4: Comparison of solutions for SSGA and NSGA-II on a small scenario: (a)–(c) one AP and (b)–(d) two APs.

power (–dBm) for each AP technology (i.e., sensitivity to environment). This stands for an additional improvement giving users a greater optimization and experimentation grade compared to other WLAN planning tools or MOGA approaches.

Moreover, we have modified NSGA-II to develop a multithread version to parallelize the effort in processing the algorithm’s objectives. In this sense, we experienced that the higher processing requirements were spent in the evaluation of the strength signal at each map location for each AP configuration. The complexity proportionally increases with the number of resulting cells because of the number of building floors, map scale, and obstacles included. To this end, WiFiSim was programmed to use up to 8 simultaneous threads (set to 4 by default).

4. Experimentation

This section shows a comparative study about the performance and scalability of the SSGA and NSGA-II approaches. To this end, three target applications were considered: small, medium, and large scenarios. Experimenting with large environments has serious concerns compared with small and medium scenarios. Larger areas have tendency to contain more living rooms and obstacles to form a more complex morphology, thus resulting in a higher division of the map cells. This implies higher search spaces to compute with significant impact on uncertainty and execution time. In order to evaluate the tests, the parameter settings for both algorithms are summarized in Tables 2 and 3.

4.1. WLAN Design on Small Scenarios. This case study consisted in fixing two wireless clients on a small symmetric scenario where one and two APs were deployed anywhere

TABLE 2: Parameter settings for SSGA.

Seed	Seed 1
Mutation type	By destruction
Selection tournament size	Minimum
Crowding	By parents
AP power	200 mW, Cisco Aironet 1100

TABLE 3: Parameter settings for NSGA-II.

Threads	4
Population size	500
Max. evaluations	5000
Integer crossover probability	90%
Binary crossover probability	90%
Integer mutation probability	3%
Binary mutation probability	3%
AP power	200 mW, standard
Protocol	IEEE 802.11g
Maximum bandwidth	54 Mbps
Minimum coverage	25%

on the map. The map was set with a scale of 1:65 meters per pixel, which is equivalent to 29.28 m². This case study served to demonstrate how NSGA-II and SSGA managed their solutions to optimally deploy the WLANs. We observed that a solution with one AP did not cover the whole map (74% in Figure 4(a) and 82% in Figure 4(c)) whereas a solution considering two APs covered the complete scenario (Figures 4(b) and 4(d)). However, the algorithms successfully found similar solutions to optimally reach the two wireless clients in their locations.

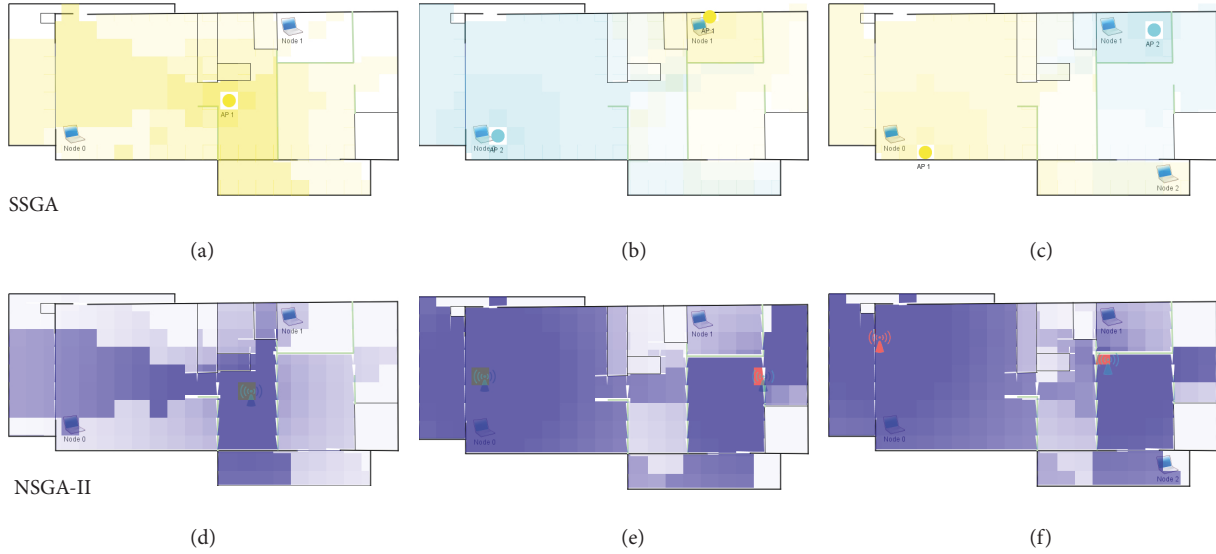


FIGURE 5: Comparison of solutions for SSGA and NSGA-II: (a)–(d) one AP covering one client, (b)–(e) two APs covering two clients, and (c)–(f) two APs covering three clients.

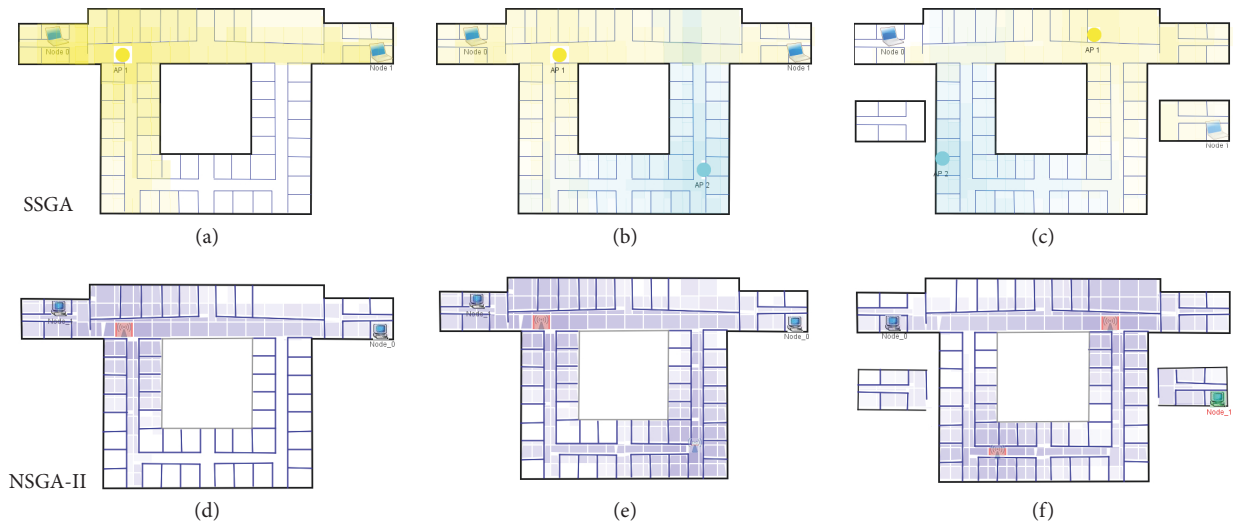


FIGURE 6: Comparison of solutions for SSGA and NSGA-II: (a)–(d) one AP for two nodes, (b)–(e) two APs for two nodes, and (c)–(f) two APs in two floors.

4.2. WLAN Design on Medium Scenarios. For this test we used a real map house with two bedrooms, two bathrooms, a dining room, a living room, and two terraces on a floor of 237.91 m^2 (Figure 5). The map was set with a scale of 1 : 32 meters per pixel. The scenario was modelled using a special contour-type material to form the outlines of the building (black thick lines with -95 dB), concrete walls for thin inner walls (green lines with -18 dB), construction beams for internal walls (black thin lines with -10 dB), a special “not evaluable” material for areas like closets (black lines with -80 dB), and a glass material for the doors accessing the terraces (blue lines with -2 dB).

The case study consisted in deploying one AP covering one client, two APs covering two clients, and two APs covering three clients. To this end, we placed the wireless

clients at the living room (node 0), a bedroom (node 1), and a terrace (node 2). Although SSGA and NSGA-II achieved similar coverage percentages on the map (between 64% and 91% for the three examples), we observed that SSGA provided a distribution more centered on the nodes than NSGA-II when the number of nodes was higher than the APs. On the contrary, NSGA-II tried to provide a more global and diverse coverage on the map (i.e., more balanced).

4.3. WLAN Design on Large Scenarios. The following experiment was conducted in a typical office environment consisting of a $73 \times 40 \text{ m}^2$ building with rooms, corridors, and open spaces with a scale of 1 : 10 meters per pixel corresponding to 2920 m^2 (Figure 6). Because of this structure, we created an exclusion zone around a landscaped courtyard at the center

TABLE 4: Performance of SSGA in a small scenario.

Target APs	Solutions found	Min. coverage	Max. coverage	AP location	Time (s)
1	1	64%	74%	Walls	0.10 ± 0.00
2	3	80%	84%	Walls	0.30 ± 0.94
3	4	80%	88%	Walls	0.36 ± 0.05
4	5	88%	92%	Walls	0.42 ± 0.04
5	6	96%	96%	Walls	0.66 ± 0.07
6	6	96%	100%	Walls	0.99 ± 0.10
7	4	100%	100%	Walls	0.82 ± 0.10
8	5	100%	100%	Walls	0.74 ± 0.05
<i>Average</i>					<i>0.54 ± 0.06</i>
1	1	64%	74%	Anywhere	0.10 ± 0.00
2	5	80%	84%	Anywhere	0.37 ± 0.08
3	3	84%	88%	Anywhere	0.32 ± 0.04
4	4	88%	92%	Anywhere	0.44 ± 0.05
5	6	92%	96%	Anywhere	0.69 ± 0.11
6	6	92%	96%	Anywhere	0.60 ± 0.08
7	6	96%	100%	Anywhere	0.62 ± 0.08
8	9	100%	100%	Anywhere	1.04 ± 0.11
<i>Average</i>					<i>0.52 ± 0.07</i>

of the building using the “not evaluable” special obstacle. The goal was to prevent the algorithms from including APs in uninteresting areas and considering only working areas to look for the optimal solutions. In addition, the environment was modelled using a contour line for the building perimeter (black lines with -95 dB), brick walls to separate the office rooms (blue lines with -3 dB), and a “roof” obstacle with 7 dB of attenuation to separate the two building floors.

The case study consisted in observing how the algorithms deployed one AP for two nodes, two APs for two nodes, and two APs in two floors located at the building walls. From the results, we obtained similar wireless coverage in the three examples (67% , 89% , and 99% , resp.). Moreover, we observed that SSGA and NSGA-II found similar AP distribution with slight differences. This suggests a great accuracy and repeatability on the algorithm solutions, thus providing a high stability regardless of the strategy chosen.

4.4. Time Analysis. In order to evaluate the complexity of the SSGA and NSGA-II strategies, we extended the above case studies with additional configurations and conducted a comprehensive analysis on the time cost. The methodology consisted in computing the average times and standard deviations of the algorithm solutions for a series of 10 records per setting. To this end, we captured the calls to the Java methods of SSGA and NSGA-II with VisualVM 1.3.9. The maps were modeled with 8, 36, and 120 building structures for the small, medium, and large scenarios, respectively, in a floor area of 800×600 pixels of resolution. For it, the experiments were performed with an Intel® Core™ i7 (2.6 GHz, 16-GB RAM).

From the results, we found in general that the SSGA executions lasted more; the higher the map resolution, the

higher the number of floor obstacles and the higher the target nodes or APs to search (Tables 4, 6, and 8). In addition, we found no evidence that the position of the APs within the map influenced the execution times (i.e., at the walls or anywhere) other than the number of solutions computed. These arguments are also valid for NSGA-II except for the execution times in medium and large environments which improved more with higher number of nodes deployed in the map (Tables 5, 7, and 9). This suggests that the uncertainty decreases as more clients are covered on the map. In other words, covering a single node requires more possible solutions of APs to be computed (i.e., more processing time) than covering several clients at the same time (i.e., more fitted solutions).

In relation to the size of the scenarios, we encountered significant differences in the execution times. In this sense, we found a better performance of NSGA-II compared with SSGA being more remarkable for the medium scenario. Moreover, it should be also mentioned that while SSGA requires one run per optimal solution searching (for one AP, two APs, etc.), NSGA-II computes a set of 32 optimal solutions in a single execution. This makes the NSGA-II approach significantly more efficient than the SSGA strategy in global terms.

With respect to the map resolutions, we attained times ranging from 0.54 ± 0.06 s to 2.25 ± 0.14 s for the small scenarios, from 27.76 ± 2.87 s to 40.06 ± 5.95 s for the medium scenarios, and from 11.98 ± 1.59 minutes to 17.15 ± 1.78 minutes for the large scenarios. This means that the algorithms are very sensitive to the map resolution, with the processing times being more suitable for small and medium scenarios but not so scalable for large environments due to the higher search spaces to compute. Nonetheless, these

TABLE 5: Performance of NSGA-II in a small scenario.

Nodes deployed	Min. APs	Min. coverage	Max. coverage	AP location	Time (s)
1	1	65%	87%	Walls	1.47 ± 0.07
2	2	100%	100%	Walls	1.54 ± 0.10
3	2	100%	100%	Walls	1.77 ± 0.05
4	2	100%	100%	Walls	2.10 ± 0.01
5	2	100%	100%	Walls	2.31 ± 0.11
6	2	100%	100%	Walls	2.54 ± 0.10
7	2	100%	100%	Walls	2.90 ± 0.10
8	2	100%	100%	Walls	3.15 ± 0.09
<i>Average</i>					<i>2.22 ± 0.07</i>
1	1	65%	87%	Anywhere	1.40 ± 0.11
2	2	100%	100%	Anywhere	1.57 ± 0.08
3	2	100%	100%	Anywhere	1.88 ± 0.09
4	2	100%	100%	Anywhere	2.13 ± 0.14
5	2	100%	100%	Anywhere	2.23 ± 0.20
6	2	100%	100%	Anywhere	2.49 ± 0.16
7	2	100%	100%	Anywhere	3.05 ± 0.13
8	2	100%	100%	Anywhere	3.31 ± 0.24
<i>Average</i>					<i>2.25 ± 0.14</i>

TABLE 6: Performance of SSGA in a medium scenario.

Target APs	Solutions found	Min. coverage	Max. coverage	AP location	Time (s)
1	1	76%	76%	Walls	0.54 ± 0.21
2	6	91%	93%	Walls	15.45 ± 1.88
3	5	91%	95%	Walls	20.07 ± 4.31
4	11	93%	98%	Walls	39.79 ± 8.93
5	8	96%	100%	Walls	36.66 ± 6.39
6	13	97%	100%	Walls	72.59 ± 8.74
7	12	98%	100%	Walls	72.39 ± 9.43
8	19	98%	100%	Walls	63.03 ± 7.72
<i>Average</i>					<i>40.06 ± 5.95</i>
1	1	76%	76%	Anywhere	0.83 ± 0.10
2	5	88%	92%	Anywhere	10.17 ± 1.20
3	5	92%	94%	Anywhere	19.88 ± 3.25
4	9	94%	97%	Anywhere	31.96 ± 4.30
5	10	95%	99%	Anywhere	36.87 ± 3.51
6	6	96%	100%	Anywhere	23.29 ± 2.06
7	12	97%	100%	Anywhere	67.79 ± 10.50
8	14	98%	100%	Anywhere	124.41 ± 22.47
<i>Average</i>					<i>39.40 ± 5.93</i>

times are in accordance with the times typically achieved by implementations based on genetic algorithms as described through [47–49]. As a solution, we enlarged the workspace of WiFiSim to reduce the resolution when simulating large scenarios. With this goal, the map resolution was changed from 1 : 10 meters by pixel in a workspace of 800×600 pixels to 1 : 20 meters by pixel in a workspace of 1600×1200 pixels.

Considering the large scenario and the worst execution times from Tables 8 and 9, we obtained a reduction from 39.08 min to 15.65 min for SSGA and from 16.37 min to 6.93 min for NSGA-II. This allows reducing the processing cost to more manageable times, thus resulting in an improvement of 40.05% and 42.34%, respectively. As a result, this test suggests that larger workspaces must be utilized to simulate larger

TABLE 7: Performance of NSGA-II in a medium scenario.

Nodes deployed	Min. APs	Min. coverage	Max. coverage	AP location	Time (s)
1	1	43%	100%	Walls	32.79 ± 2.94
2	2	56%	100%	Walls	29.87 ± 2.46
3	2	59%	100%	Walls	30.11 ± 2.31
4	3	54%	100%	Walls	25.29 ± 2.26
5	3	57%	100%	Walls	24.69 ± 3.01
6	3	53%	100%	Walls	26.39 ± 3.18
7	3	57%	100%	Walls	27.64 ± 3.07
8	3	65%	100%	Walls	27.68 ± 1.70
<i>Average</i>					28.05 ± 2.62
1	1	42%	100%	Anywhere	30.31 ± 3.17
2	2	63%	100%	Anywhere	33.44 ± 1.47
3	2	61%	100%	Anywhere	30.16 ± 2.40
4	3	63%	100%	Anywhere	24.74 ± 2.91
5	4	72%	100%	Anywhere	24.61 ± 2.72
6	4	70%	100%	Anywhere	24.11 ± 2.48
7	4	72%	100%	Anywhere	26.95 ± 2.86
8	4	75%	100%	Anywhere	27.83 ± 4.89
<i>Average</i>					27.76 ± 2.87

TABLE 8: Performance of SSGA in a large scenario.

Target APs	Solutions found	Min. coverage	Max. coverage	AP location	Time (min)
1	1	67%	67%	Walls	0.45 ± 0.06
2	7	87%	95%	Walls	5.05 ± 0.79
3	6	94%	98%	Walls	5.32 ± 0.55
4	6	97%	99%	Walls	7.34 ± 0.64
5	9	99%	100%	Walls	15.59 ± 0.90
6	6	100%	100%	Walls	17.80 ± 3.03
7	12	100%	100%	Walls	29.06 ± 3.49
8	9	100%	100%	Walls	18.96 ± 2.00
<i>Average</i>					12.45 ± 1.43
1	1	64%	64%	Anywhere	0.74 ± 0.13
2	1	94%	94%	Anywhere	3.14 ± 0.50
3	3	98%	98%	Anywhere	4.21 ± 0.80
4	7	99%	100%	Anywhere	11.79 ± 1.20
5	8	99%	100%	Anywhere	18.59 ± 2.38
6	11	100%	100%	Anywhere	22.23 ± 0.29
7	17	100%	100%	Anywhere	37.40 ± 5.82
8	18	100%	100%	Anywhere	39.08 ± 3.13
<i>Average</i>					17.15 ± 1.78

environments, being besides more appropriate when drawing small details in large scenarios due to the larger view.

5. Conclusions and Future Works

WLAN design and planning is a complex task that demands of engineers high analytical and troubleshooting skills to

offer the best network performance and usage. While hand-operated traditional methodologies are not effective for practitioners due to cost and time, computer-aided systems must provide a high degree of realism and modelling capabilities to be useful. To contribute to this field, this paper presented an engineering tool—called WiFiSim—developed to facilitate the automatic WLAN planning of complex environments

TABLE 9: Performance of NSGA-II in a large scenario.

Nodes deployed	Min. APs	Min. coverage	Max. coverage	AP location	Time (min)
1	3	33%	99%	Walls	15.79 ± 1.12
2	3	32%	99%	Walls	14.55 ± 0.09
3	3	29%	99%	Walls	14.48 ± 1.52
4	4	39%	98%	Walls	12.63 ± 0.57
5	6	50%	99%	Walls	13.62 ± 1.85
6	6	60%	99%	Walls	10.80 ± 0.74
7	8	62%	100%	Walls	10.14 ± 0.62
8	8	65%	100%	Walls	10.40 ± 0.96
<i>Average</i>					<i>12.80 ± 0.94</i>
1	3	35%	99%	Anywhere	16.37 ± 1.36
2	3	32%	99%	Anywhere	11.93 ± 2.31
3	3	27%	99%	Anywhere	13.07 ± 1.89
4	5	47%	99%	Anywhere	10.83 ± 1.22
5	5	44%	99%	Anywhere	10.39 ± 1.46
6	6	46%	99%	Anywhere	9.85 ± 1.86
7	7	60%	99%	Anywhere	12.65 ± 0.52
8	6	55%	100%	Anywhere	10.66 ± 2.07
<i>Average</i>					<i>11.98 ± 1.59</i>

and to assist in decision-making prior to real deployment. This tool that affords the complete behavior modeling of Wi-Fi networks in Layer 1 and 2 of the OSI model has been improved with an optimization algorithm based on evolutionary genetic. The goal was to procure a set of suboptimal solutions very close to the best for the AP location problem considering the largest covered area and maximum strength signal as decision criteria.

With this purpose, a previous approach based on a monoobjective genetic algorithm (i.e., SSGA) was extended with a multiobjective genetic algorithm (i.e., NSGA-II). Thus, in addition to obtaining several AP locations for up to 8 different selectable configurations, the main advantage of using NSGA-II was the capability to attain 32 solutions at once with several simultaneous configurations of APs. This approach was also designed to satisfy the user design constraints concerning the transmission rate and Wi-Fi technology used in the WLAN.

To assess both approaches and highlight the advantages of the second algorithm, several scenarios with small, medium, and large areas were used to simulate typical Wi-Fi environments embodying an office, a house, and a campus. The aim of the various tests was to evaluate the consistency of the algorithms—and how they computed their optimal solutions—along different scenarios and map resolutions in relation to the structure complexity and the search spaces. From the results, we encountered that the algorithm executions lasted more; the higher the map resolution, the higher the number of map obstacles and the higher the number of targets (i.e., nodes or APs). Moreover, we found no significant difference in the execution times due to the position of the APs within the map (i.e., at the walls or anywhere). Concerning the map size, the algorithms achieved very affordable

processing times for small and medium scenarios, not so scalable for large environments due to the higher search spaces. Although the times were in accordance with the typical requirements for GA-based implementations, the execution cost can be reduced to more manageable times (up to 40.05% and 42.34%) using larger workspaces to model larger environments (e.g., from a map resolution of 1:10 meters by pixel in a workspace of 800 × 600 pixels to 1:20 meters by pixel in a workspace of 1600 × 1200 pixels). As a result, we conclude that the NSGA-II approach achieved higher performance in general terms compared with the SSGA strategy.

Regarding the present, we are currently working on connecting WiFiSim with an eLearning tool for the programming, study, and distribution of wireless protocols based on an institutional web repository. In addition, future developments are focused on improving several functional and technical capabilities of WiFiSim. In this sense, we are working to extend the features on the PHY and MAC layers to enhance the software realism. Regarding the PHY layer, this comprises new signal measurements given by the BER, EIRP, and RSSI, an antennae modeling, support for IEEE 802.11ac/ad, larger Wi-Fi vendor library, and the cost of the resulting infrastructures. As for the MAC layer, this consists in providing the complete IEEE 802.11 frame format, including new frames as the RIFS, EIFS, and PS-Poll frames, new connection procedures as the association, reassociation, authentication, and deauthentication, and the protection mode for mixed scenarios with DSSS and OFDM, as well as power management by the TIM and DTIM mechanisms.

Finally, a set of selected videos and the modelling simulation software are offered for the free use and evaluation at <http://www.uhu.es/tomas.mateo/wifisim>.

Abbreviations

AP:	Access point
BER:	Bit error rate
CA:	Collision avoidance
CSMA:	Carrier sense multiple access
CTS:	Clear to send
DSSS:	Direct sequence spread spectrum
DTIM:	Delivery traffic indication map
EIFS:	Extended interframe space
EIRP:	Effective isotropic radiated power
FSPL:	Free space path loss
GA:	Genetic algorithm
GUI:	Graphical user interface
IEEE:	Institute of Electrical & Electronics Engineers
IAGA:	Improved adaptive genetic algorithm
ICT:	Information and communication technologies
LAN:	Local area network
MAC:	Media access control
MICRO-GA:	Microgenetic algorithm
MOGA:	Multiobjective genetic algorithm
NPGA:	Niched Pareto genetic algorithm
NPGA2:	Improved niched Pareto genetic algorithm
NSGA-II:	Nondominated sorting genetic algorithm II
OFDM:	Orthogonal frequency division multiplexing
OSI:	Open system interconnection
PAES:	Pareto archived evolution strategy
PESA:	Pareto envelope-based selection algorithm
PHY:	Physical
PS-POLL:	Power save poll
RF:	Radio frequency
RIFS:	Reduced interframe space
RSSI:	Received signal strength indicator
RTS:	Request to send
RW:	Random walk
SA:	Simulated annealing
SNR:	Signal to noise ratio
SPEA:	Strength Pareto evolutionary algorithm
SSGA:	Steady-state genetic algorithm
TIM:	Traffic indication map
TS:	Tabu search
WI-FI:	Wireless Fidelity
WIFISIM:	Wireless Fidelity Simulator
WLAN:	Wireless area network.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

The authors would like to express their very great appreciation to F. A. Márquez Hernández, D. Ortiz Fuentes, B. Moriña Arias, and M. E. Pedraza Escudero for their valuable and constructive work that helped to improve this research.

References

- [1] J. Kennington, E. Olinick, and D. Rajan, "Introduction to optimization in wireless networks," *International Series in Operations Research and Management Science*, vol. 158, pp. 1–6, 2011.
- [2] D. Hucaby, *CCNA Wireless 640-722 Official Cert Guide*, Cisco Press, 2014.
- [3] D. Vassis and G. Kormentzas, "Performance analysis of IEEE 802.11 ad hoc networks in the presence of hidden terminals," *Computer Networks*, vol. 51, no. 9, pp. 2345–2352, 2007.
- [4] T. D. J. Mateo Sanguino and F. A. Márquez, "Optimal design and deployment of wireless LANs based on evolutionary genetic strategy," *Advances in Intelligent Systems and Computing*, vol. 474, pp. 283–291, 2016.
- [5] T. J. Mateo Sanguino, F. A. Márquez Hernández, and C. Serrano López, "Evaluating a computer-based simulation program to support wireless network fundamentals," *Computers & Education*, vol. 70, pp. 233–244, 2014.
- [6] T. De J. Mateo Sanguino, C. S. Lopez, and F. A. M. Hernandez, "WiFiSiM: An educational tool for the study and design of wireless networks," *IEEE Transactions on Education*, vol. 56, no. 2, pp. 149–155, 2013.
- [7] S. Vasudevan, *A Simulator for analyzing the throughput of IEEE 802.11b Wireless LAN Systems [Master, thesis]*, Virginia Polytechnic Institute and State University, 2005.
- [8] C. Karthik and J. Paul, "Optimal design of wireless local area networks (WLANs) using simulation," in *Proceedings of the 2009 IEEE Military Communications Conference, MILCOM 2009, USA, October 2009*.
- [9] S. U. Nnebe, G. N. Onoh, and C. O. Ohaneme, "Empirical Analysis of Signal-to-Interference Ratio Variations in IEEE 802.11b WLAN," *International Journal of Innovative Research in Science, Engineering and Technology*, vol. 1, no. 2, pp. 263–270, 2012.
- [10] W. Zhu, F. Gao, Z. Song, and K. He, "Theoretical Analysis and Test of EMF in TDFI Bus," in *Progress in Electromagnetics Research Symposium Abstracts*, Guangzhou, China, 2014.
- [11] J. Gu, J. Zheng, and S. Zhang, "A graphical user interface in WLAN monitoring and management system," *Journal of Networks*, vol. 7, no. 4, pp. 644–651, 2012.
- [12] E. Neha Sharma and G. C. Lall, "Enhancement and Characterization of Indoor Propagation Models," *International Journal of Scientific & Engineering Research*, vol. 2, no. 11, pp. 1–7, 2011.
- [13] I. Aldasouqi and W. A. Salameh, "Using GIS in Designing and Deploying Wireless Network in City Plans," *International Journal of Computer Networks*, vol. 6, no. 4, pp. 66–75, 2014.
- [14] J. Scheuner, G. Mazlami, and D. Schöni, "Probr - a generic and passive WiFi tracking system," in *Proceedings of the 2016 IEEE 41st Conference on Local Computer Networks (LCN)*, pp. 495–502, IEEE, Dubai, United Arab Emirates, November 2016.
- [15] "Wolf WiFi VIRI Help Documentation," Tech. Rep., 2011, <https://twitter.com/wolfwifi>.
- [16] J. Hunt, "Working successful motion control via standard Ethernet," *The Journal of Industrial Network Connectivity*, vol. 71, pp. 33–38, 2012.
- [17] S. Vigneshwaran and C. Selvaraj, "Implementation of WLAN N and Estimation of Co Channel and Adjacent Channel Interference," *Journal of Computer Engineering Information Technology*, vol. 5, no. 2, 2016.
- [18] S. Blomqvist, *Improvement Proposal for Wireless Office Networks [BSc Degree]*, Mälardalen University, 2017.

- [19] R. J. Bartz, *Mobile Computing Deployment and Management: Real World Skills for CompTIA Mobility+ Certification and Beyond*, Wiley & Sons, 2015.
- [20] L. Phifer, "Managing WLAN Risks with Vulnerability Assessment," in *White Paper*, Core Competence, Inc., 2008.
- [21] J. Kruys and L. Qian, "RF Spectrum, Usage and Sharing," in *Sharing RF Spectrum with Commodity Wireless Technologies*, Signals and Communication Technology, pp. 3–14, Springer Netherlands, Dordrecht, 2011.
- [22] T. Vanhatupa, "Wi-Fi Capacity Analysis for 802.11ac and 802.11n - Theory & Practice," in *White Paper*, Ekahau, Inc., 2013.
- [23] "Optimus Information, Inc. Open-Source vs. Proprietary Software Pros and Cons," in *White paper*, 2015.
- [24] K. M. Reineck, *Evaluation and Comparison of Network Simulation Tools [Master, thesis]*, University of Applied Sciences, Bonn-Rhein-Sieg and Fraunhofer-Gesellschaft, 2008.
- [25] R. M. Whitaker and S. Hurley, "Evolution of planning for wireless communication systems," in *Proceedings of the 36th Annual Hawaii International Conference on System Sciences, HICSS 2003*, USA, January 2003.
- [26] A. Mc Gibney, M. Klepal, and D. Pesch, "Agent-based optimization for large scale WLAN design," *IEEE Transactions on Evolutionary Computation*, vol. 15, no. 4, pp. 470–486, 2011.
- [27] A. Mc Gibney, M. Klepal, and D. Pesch, "A Wireless Local Area Network Modeling Tool for Scalable Indoor Access Point Placement Optimization," in *Proceedings of the 2010 Spring Simulation Multiconference*, vol. 163, pp. 1–8, 2010.
- [28] M. Unbehaun and M. Kamenetsky, "On the Deployment of Picocellular Wireless Infrastructure," *IEEE Wireless Communications Magazine*, vol. 10, no. 6, pp. 70–80, 2003.
- [29] S. A. Chen, Y. H. Lee, R. Y. Yen et al., "Optimal prediction tool for wireless LAN using genetic algorithm and neural network concept," in *Proceedings of APCC/OECC'99 - 5th Asia Pacific Conference on Communications/4th Optoelectronics and Communications Conference*, pp. 786–789 vol.1, Beijing, China, October 1999.
- [30] R.-H. Wu, Y.-H. Lee, and S.-A. Chen, "Planning system for indoor wireless network," *IEEE Transactions on Consumer Electronics*, vol. 47, no. 1, pp. 73–79, 2001.
- [31] X. Huang, U. Behr, and W. Wiesbeck, "Automatic cell planning for a low-cost and spectrum efficient wireless network," in *Proceedings of the IEEE Global Telecommunication Conference (GLOBECOM'00)*, pp. 276–282.
- [32] S. Liu, C. Zhong, L. Zhang, and L. Zhang, "Optimization design of wireless local area network based on improved adaptive genetic algorithm," in *Proceedings of the 6th World Congress on Intelligent Control and Automation, WCICA 2006*, pp. 3357–3360, China, June 2006.
- [33] K. Maksuriwong, V. Varavithya, and N. Chaiyaratana, "Wireless LAN access point placement using a multi-objective genetic algorithm," in *Proceedings of the System Security and Assurance*, pp. 1944–1949, usa, October 2003.
- [34] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," *IEEE Transactions on Evolutionary Computation*, vol. 6, no. 2, pp. 182–197, 2002.
- [35] J. Horn, N. Nafpliotis, and D. E. Goldberg, "A niched Pareto genetic algorithm for multiobjective optimization," in *Proceedings of the 1st IEEE Conference on Evolutionary Computation*, pp. 82–87, June 1994.
- [36] N. Srinivas and K. Deb, "Multiobjective function optimization using nondominated sorting genetic algorithms," *Evolutionary Computation*, vol. 2, pp. 221–248, 1994.
- [37] E. Zitzler and L. Thiele, "Multiobjective evolutionary algorithms: A comparative case study and the strength Pareto approach," *IEEE Transactions on Evolutionary Computation*, vol. 3, no. 4, pp. 257–271, 1999.
- [38] J. D. Knowles and D. W. Corne, "Approximating the nondominated front using the Pareto archived evolution strategy," *Evolutionary Computation*, vol. 8, no. 2, pp. 149–172, 2000.
- [39] D. W. Corne, J. D. Knowles, and M. J. Oates, "The Pareto-envelope based selection algorithm for multiobjective optimization," in *Parallel Problem Solving from Nature PPSN VI*, vol. 1917 of *Lecture Notes in Computer Science*, pp. 869–878, Springer, New York, NY, USA, 2000.
- [40] C. A. C. Coello and G. T. Pulido, "A micro-genetic algorithm for multiobjective optimization," in *Evolutionary Multi-Criterion Optimization: First International Conference, EMO 2001 Zurich, Switzerland, March 7–9, 2001 Proceedings*, vol. 1993 of *Lecture Notes in Computer Science*, pp. 126–140, Springer, Berlin, Germany, 2001.
- [41] M. Erickson, A. Mayer, and J. Horn, "The Niched Pareto Genetic Algorithm 2 Applied to the Design of Groundwater Remediation Systems," in *Evolutionary Multi-Criterion Optimization*, vol. 1993 of *Lecture Notes in Computer Science*, pp. 681–695, Springer Berlin Heidelberg, Berlin, Heidelberg, 2001.
- [42] D. Corne, N. Jerram, J. Knowles, and M. Oates, "PESA-II: Region Based Selection in Evolutionary Multiobjective Optimization," in *Proceedings of the Genetic and Evolutionary Computation Conference*, pp. 283–290, 2001.
- [43] E. Zitzler, M. Laumanns, and L. Thiele, "SPEA2: Improving the Strength Pareto Evolutionary Algorithm for Multiobjective Optimization," in *Proceedings of the Evolutionary Methods for Design, Optimization and Control with Applications to Industrial Problems*, 2001.
- [44] C. Pei, Z. Wang, Y. Zhao et al., "Why it takes so long to connect to a WiFi access point," in *Proceedings of the IEEE INFOCOM 2017 - IEEE Conference on Computer Communications*, pp. 1–9, Atlanta, GA, USA, May 2017.
- [45] J. J. Durillo, A. J. Nebro, and E. Alba, "The jMetal framework for multi-objective optimization: design and architecture," in *Proceedings of the IEEE Congress on Evolutionary Computation (CEC '10)*, pp. 4138–4325, July 2010.
- [46] A. A. Márquez Hernández, *Cooperación entre el Mecanismo de Inferencia y la Base de Reglas usando Algoritmos Genéticos Multiobjetivo [Ph.D. thesis]*, University of Huelva, 2008.
- [47] D. Noever and S. Baskaran, *Steady-State vs. Generational Genetic Algorithms: A Comparison of Time Complexity and Convergence Properties*, Santa Fe Institute Working Papers, 1992.
- [48] B. M. Kim, Y. B. Kim, and C. H. Oh, "A Study on the Convergence of Genetic Algorithms," *Computers & Industrial Engineering*, vol. 33, no. 3–4, pp. 581–588, 1997.
- [49] J. Seo, E. Ko, and Y. Kim, "Performance Comparison of GPUs with a Genetic Algorithm based on CUDA," in *Proceedings of the Database 2014*, pp. 36–40.




Hindawi

Submit your manuscripts at
www.hindawi.com

