UNIVERSIDADE FEDERAL DO PARÁ
INSTITUTO DE TECNOLOGIA
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA ELÉTRICA

Intelligent Positioning of Drones via Metaheuristic Optimization Algorithms for Maximizing Signal Coverage Area in Forested Environments

FLÁVIO HENRY CUNHA DA SILVA FERREIRA

DM 05/2022

UFPA / ITEC / PPGEE
Campus Universitário do Guamá
Belém-Pará-Brasil
2022

UNIVERSIDADE FEDERAL DO PARÁ
INSTITUTO DE TECNOLOGIA
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA ELÉTRICA

FLÁVIO HENRY CUNHA DA SILVA FERREIRA

Intelligent Positioning of Drones via Metaheuristic Optimization Algorithms for Maximizing Signal Coverage Area in Forested Environments

DM 05/2022

UFPA / ITEC / PPGEE
Campus Universitário do Guamá
Belém-Pará-Brasil
2022

UNIVERSIDADE FEDERAL DO PARÁ
INSTITUTO DE TECNOLOGIA
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA ELÉTRICA

FLÁVIO HENRY CUNHA DA SILVA FERREIRA

Intelligent Positioning of Drones via Metaheuristic Optimization Algorithms for Maximizing Signal Coverage Area in Forested Environments

MsC thesis submitted to the Examining Board of the Postgraduate Program in Electrical Engineering at UFPA, in order to obtain the Master's Degree in Electrical Engineering in the field of Applied Computing.

Advisor: Jasmine Priscyla Leite de Araújo
Co-Advisor: Miércio Cardoso de Alcântara Neto

UFPA / ITEC / PPGEE
Campus Universitário do Guamá
Belém-Pará-Brasil
2022

Dados Internacionais de Catalogação na Publicação (CIP) de acordo com ISBD
Sistema de Bibliotecas da Universidade Federal do Pará
Gerada automaticamente pelo módulo Ficat, mediante os dados fornecidos pelo(a) autor(a)

C972i Cunha da Silva Ferreira, Flávio Henry.
Intelligent Positioning of Drones via Metaheuristic Optimization Algorithms for Maximizing Signal Coverage Area in Forested Environments / Flávio Henry Cunha da Silva Ferreira. — 2022.
73 f. : il. color.

Orientador(a): Profª. Dra. Jasmine Priscyla Leite de Araújo
Coorientador(a): Prof. Dr. Miércio Cardoso de Alcântara Neto
Dissertação (Mestrado) - Universidade Federal do Pará, Instituto de Tecnologia, Programa de Pós-Graduação em Engenharia Elétrica, Belém, 2022.

1. IoT. 2. LPWAN. 3. Computação bioinspirada. 4. UAV. 5. Array de drones. I. Título.

CDD 621.3

UNIVERSIDADE FEDERAL DO PARÁ
INSTITUTO DE TECNOLOGIA
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA ELÉTRICA

Intelligent Positioning of Drones via Metaheuristic Optimization Algorithms for Maximizing Signal Coverage Area in Forested Environments

AUTOR: **FLÁVIO HENRY CUNHA DA SILVA FERREIRA**

DISSERTAÇÃO DE MESTRADO SUBMETIDA À AVALIAÇÃO DA BANCA EXAMINADORA APROVADA PELO COLEGIADO DO PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA ELÉTRICA DA UNIVERSIDADE FEDERAL DO PARÁ E JULGADA ADEQUADA PARA A OBTENÇÃO DO GRAU DE MESTRE EM ENGENHARIA ELÉTRICA NA ÁREA DE TELECOMUNICAÇÕES.

APROVADA EM:

BANCA EXAMINADORA:

Prof. Dr. Jasmine Priscyla Leite de Araújo
(Orientador - PPGEE/UFPA)

Prof. Dr. Miércio Cardoso de Alcântara Neto
(Avaliador Interno - PPGEE/UFPA)

Prof. Dr. Fábricio José Brito Barros
(Avaliador Interno - PPGEE/UFPA)

Prof. Dr. Gervásio Protásio dos Santos Cavalcante
(Avaliador Interno - PPGEE/UFPA)

Prof. Dr. Alfredo Gomes Neto
(Avaliador Externo ao Programa - GTEMA/IFPB)

VISTO:

Prof. Dr. Carlos Tavares da Costa Júnior
(Coordenador do PPGEE/UFPA)

Acknowledgements

First and foremost, I cannot express how much gratitude I have towards my mother Cristina and my proximate family: aunt Andréa, grandma Zuleide and my brother Daniel. They helped me immensely through my studies, my anxieties and complications, and had to be very understanding and patient at times. I hope I can make them proud. I love you all so much.

I would like to thank my advisors Jasmine and Miércio. Last year was a time where many personal tribulations affected me, and COVID-19 hindered my progress twice, but they have been supportive all the way through. Without their belief in me to drive me forward, I fear I would not be able to finish this work.

My father Flávio also gave me emotional support when I needed most, and inspired me to expand my boundaries, seek independence and to work not only for myself but to help the ones I love. After all, these are the most important things in life, and I carry these teachings to heart. I love you too, dad.

I am grateful for my best friends and how they positively influenced me over the years. Despite the epidemic, we have kept almost daily contact and helped me de-stress many times. I won't spell any names, they know exactly who they are. I wish all the best for them.

Also, I am duly thankful for the financial support received by CAPES and PPGEE-UFPA to be able to conclude my studies in graduation.

"Have no fear of perfection - you'll never reach it."
(Salvador Dalì)

Resumo

Esta dissertação tem como objetivo propor uma abordagem metaheurística para otimização de arrays de drones aplicada à maximização da área de cobertura de sistemas de comunicação sem fio, contendo veículos aéreos não tripulados (UAV, em inglês) como estações-base. Para tanto, foram analisados dois tipos de redes que utilizam UAVs: uma rede Wi-Fi padrão operando a 2,4 GHz e uma rede wireless de baixa potência (LPWAN), ambas considerando ambientes medianamente ou altamente arborizados. LPWAN são sistemas projetados para trabalhar com taxas de dados baixas que mantêm, ou até mesmo melhoram, a ampla áre de cobertura fornecida por redes de alta potência. O tipo de LPWAN escolhido para estudo é o LoRa, que opera em um espectro não licenciado de 915 MHz e requer que os usuários se conectem a gateways para transmitir informações a um servidor central - neste caso, cada drone no array tem um módulo LoRa instalado e serve como um gateway não-fixo. A fim de classificar e otimizar o melhor posicionamento para cada UAV no array, três métodos concomitantes de otimização bioinspirada foram escolhidos: o busca cuco (CS), o algoritmo de polinização de flores (FPA) e o algoritmo de ecolocalização de morcegos (BA). Os métodos têm uma distribuição de espaço de busca baseada em voos de Lévy / Mantegna (CS, FPA) e distribuição normal (BA) e apresentam resultados de desempenho distintos para ambos os casos de rede de drones. Os resultados da otimização de posicionamento são então simulados e apresentados via MATLAB, primeiro para a rede Wi-Fi e depois para uma rede IoT-LoRa. Além disso, um modelo de propagação ajustado empiricamente com medidas realizadas no campus da UFPA foi desenvolvido para obter um modelo de propagação em ambientes com florestas. Por fim, o posicionamento dos drones utilizando o modelo de propagação ajustado com medidas é comparado com o posicionamento utilizando o modelo teórico clássico, mostrando que o modelo ajustado é mais eficiente na representação do ambiente com florestas do que o modelo clássico usualmente utilizado em publicações recentes.

Palavras-chave: IoT; LPWAN; LoRA; Metaheurística; Computação Bioinspirada; UAV; Array de Drones.

Abstract

This dissertation aims to provide a metaheuristic approach to drone array optimization applied to coverage area maximization of wireless communication systems, with unmanned aerial vehicle (UAV) base-stations. For this purpose, two types of networks utilizing UAVs have been analyzed: a standard Wi-Fi network operating at 2.4 GHz, and a low-power wireless area network (LPWAN), both considering medium to high-density forest environments. LPWAN are systems designed to work with low data rates but still keep, or even enhance, the extensive area coverage provided by high-powered networks. The type of LPWAN chosen herein is LoRa, which operates at an unlicensed spectrum of 915 MHz, and requires users to connect to gateways in order to relay information to a central server – in this case, each drone in the array has a LoRa module installed to serve as a non-fixated gateway. In order to classify and optimize the best positioning for every UAV in the array, three concomitant bioinspired optimization methods have been chosen: the cuckoo search (CS), the flower pollination algorithm (FPA) and the bat echolocation algorithm (BA). All of these methods have a search space distribution based on Lévy / Mantegna flights (CS, FPA) and normal distribution (BA), and present distinct performance results for both drone array network cases. Positioning optimization results are then simulated and presented via MATLAB, first for the Wi-Fi network and later for a high-range IoT-LoRa network. An empirically adjusted propagation model with measurements carried out on the UFPA campus was developed to obtain a propagation model in forested environments. Finally, drone positioning utilizing the propagation model corrected with measurements is compared with the positioning using the classical theoretical model, showing that the corrected model is more efficient in representing the forest environment than the classical model usually used in recent publications.

Keywords: IoT; LPWAN; LoRa; Metaheuristics; Bioinspired Computing; UAV; Drone Array.

Summary

List of Figures

List of Tables

List of Abbreviations

3GPP	3rd Generation Partnership Project
4G-LTE	4th Generation of Wireless Communications - Long Term Evolution
5G	5th Generation of Wireless Communications
BA	Bat Algorithm
BW	Bandwidth
CEAMAZON	Center of Excellency in Energetic Efficiency of Amazon
CS	Cuckoo Search
DE	Differential Evolution
DSC	Drone-Station Controller
DSC	Drone-Station Controller
IoT	Internet of Things
FPA	Flower Pollination Algorithm
GA	Genetic Algorithm
GPS	Global Positioning System
IoT	Internet of Things
LCT-UFPA	Computation and Telecommunications Lab - UFPA
LDPL	Log-Distance Path Loss Model
LPWAN	Low-Power Wide Area Network
LTE-M	Long Term Evolution - Machine Type Communication
LTE	Long Term Evolution
LoRaWAN	Long Range Wide Area Network
LoRa	Long-Range

LoS	Line-Of-Sight
NB-IoT	Narrow Band – Internet of Things
NFC	Near-Field Communication
NLoS	Non-Line-of-Sight
PLE	Path Loss Exponent
PSO	Particle Swarm Optimization
P_r	Received Power
RMSE	Root Mean Squared Error
SA	Simulated Annealed
SF	Spreading Factor
SINR	Signal-Interference-Plus-Noise Ratio
Tx, Rx	Transmit, Receive
UAV	Unmanned Aerial Vehicle
UFPA	Federal University of Pará / Universidade Federal do Pará
UNB	Ultra Narrow Band
WAN	Wide-Area Network
WSN	Wireless Sensor Network

1 Introduction

Wireless communications have an intricate relationship with the technologies of Internet of Things (IoT), a fairly new concept made for uniting and connecting all sorts of devices, services and commodities in the contemporary world. Many of such wireless data-transmitting protocols associated with IoT are high-speed networks to favor the heavy data usage of mobile users, such as wide-area networks (WAN) and 4G-LTE or 5G systems. However, it is not always that these high-power, high data rate services are needed. Often, devices with low battery consumption cannot bear to operate at such power-consuming paces, which characterizes a need for another kind of network: the low-power wide-area networks (LPWAN).

LPWAN are systems designed to work with low data rates but still keep the extensive area coverage provided by high-powered networks. Thus, a private and secure network for the connection of low-power devices can be established, keeping financial costs and battery consumption at the minimal. For instance, wireless sensor networks, or WSN, are known to function effectively to models of different types of LPWAN, as seen in (BORGES; VELEZ; LEBRES, 2014). One of these such types is Long-Range, also known as LoRa, which is, currently, extensively studied and the most readily available.

In (LAVRIC; POPA, 2017), the authors discourse over the functioning of LoRa, its benefits and its challenges. Given that is a bidirectional manner of sending data between users and transmitters, a LoRaWAN can easily export information such as geolocation, signal-interference-plus-noise ratio (SINR) and received signal strength indicators (RSSI). This kind of flexibility and inherent monitoring of the network makes the integration between receiving sensors and transmitting drones a promising implementation.

As LoRa modules are lightweight and generally have their own separate battery, they can easily be attached to UAVs without interfering with its flight capabilities or power consumption. Given that the battery life of drones is generally low, due to their high-power flight demands, LoRa is an economic alternative to make drone-array networks work.

For this dissertation, the focus is set on how to optimize an array of UAVs, utilizing as examples a 2.4 GHz Wi-Fi system and a LoRaWAN, to give the best coverage area possible, in which all network users/sensors are satisfactorily connected. In order to accomplish this coverage area maximization, some metaheuristic optimization techniques are to be used. These will also determine which details each sensor will use to connect to the drone base-stations, so they can obtain maximum coverage or maximum data rate.

Therefore, a simulation was constructed in software MATLAB, simulating a forest environment, with the purpose of giving results that shall be applicable for the region in which the measured data was collected.

1.0.1 Motivations

The greater motivations for the study are the lack of drone-array positioning studies for forest environments, and the prospects of utilizing this in conjunction to computational optimization to obtain satisfactory data coverage results in medium to densely-forested ambiences. Furthermore, the use of drone-arrays to provide LPWAN connections to sensors located on the ground, and in this kind of environment, highlights the work herein as novel in the literature.

1.0.2 General Goals

The general goals of the study are to create an optimization environment for maximizing the area of data coverage for drone base-station arrays in medium to densely-forested environments, and test it with simulations for the frequencies of Wi-Fi (2.4 GHz) and LoRa (915 MHz). It is important to denote that, in this study, only the LoRa application possesses measured data feedback, with Wi-Fi simulation being purely theoretical.

1.0.3 Specific Goals

This Master's Thesis has the following specific and secondary goals:

- Utilize of measured SINR, SF, RSSI and geolocation values provided by the LoRa sensors in order to simulate the positioning of drones within the array;
- Produce the best results for maximum area coverage in which all users are receiving acceptable signal quality and where the fewest possible drones are to be in operation;
- Prove, by employing a propagation model utilizing measured data, that the simulation can foresee and apply the effects of SF on area coverage in a satisfactory manner;
- Present simulation hypotheses in which the methodology and metaheuristic optimization techniques are put to test, in order to produce reliable results and prove their validity.

1.0.4 Contributions

The main contributions for which this study aims are thus:

- A simulation environment in which drone base-stations are able to connect and serve a high number of users for Wi-Fi internet usage, considering a search space with forested characteristics, to maximize data coverage areas whilst maintaining satisfactory SINR values;
- A simulation environment in which drone base-stations can connect several sensors in a flexible way, with varying SF spectra, to reach optimal data coverage areas according to user demand;
- The comparison of performance between a classic drone propagation model to an empirical model measured for LoRa frequencies and applications, in a forested environment.

1.0.5 Structure of the Dissertation

There are seven chapters in this master degree's thesis, organized as such:

- Chapter 2 denotes the main works currently in the literature and state-of-the-art on drone-array base station systems, as well as optimization of data coverage by area;
- Chapter 3 discourses on the theory of LPWAN and LoRa communications, its differences to high-power wireless networks and its particularities in relation to the work present herein;
- Chapter 4 explains the theory behind the bioinspired computational optimization methods utilized in the study;
- Chapter 5 shows the methodology of the propagation models utilized in the simulation, the conducting of the measurement campaigns, the proposal of a propagation model that ties up with measured data and, lastly, the structure of the simulation algorithms;
- Chapter 6 presents the results of the simulations, for both the frequencies of Wi-Fi and LoRa separately, and its comparison to measured data for LoRa applications;
- And Chapter 7 comments said results and considerations of the dissertation and serves as its conclusion.

2 Correlated Works

2.1 Initial Considerations

In this chapter, there will be an exposure of works in the literature that are relevant for the study made in the dissertation. It shall be separated in the following categories: LoRa implementations with UAVs, drone-array optimization via metaheuristics and studies with UAV systems in forested environments.

2.2 LoRa Implementations with UAVs

In (CARUSO et al., 2021), a model for a LoRa system of data collecting for precision agriculture is proposed. The aim of this study is to perceive how close must the drone in this type of setting fly by the sensors in order to collect data within a given quality threshold. It considers the pathing of only one drone in a plantation grid which has various sensors separated from each other by a certain distance k. Some metrics are considered in the study for both small and large plantation fields, such as the minimum autonomy time for the drone, the packet transmission times for different SF and the maximum number of sensors that a drone can attend for different SF.

For the study in (DAMBAL et al., 2019), there is an analysis of LoRa networks for urban and suburban environments. Given that the reach of the connection link can be diminished if there are many shadowing objects and effects in its path, the authors propose an air-to-ground LoRaWAN operating in outdoor settings (the third scenario in the paper). Measurements on the received signal strength have been conducted for both heights of 25 meters and 50 meters above ground, as well as for vertical (V-V) and cross-polarization (V-H) of transmitting and receiving antennas. The height of 50 meters with vertical polarization demonstrates the best results.

In (DELAFONTAINE et al., 2020), a UAV is used to better estimate the location of a certain LoRa beaconing node in the ground. This optimization is made possible by utilizing the received power formula, and then obtaining the distance from it. A simple greedy algorithm is then applied to compare the geolocation data that the received power formula has suggested, and iterates in order to find a very proximate real location result. The authors claim that this method can increase location precision by up to ten times compared to the one found in fixed LoRa gateways.

An interesting approach to the usage of a LoRaWAN is found in (RAHMADHANI et al., 2018), where it is utilized as a secondary (or emergency) communication network for drone package delivery and telemetry. This is made possible by a LoRa module coupled to the drone and a fixed gateway transmitting at around 2.4 GHz, that is, the Wi-Fi spectrum. Telemetry data, in this paper, denote results almost in real-time, with a minimum interval of up to three seconds and only 10 to 20 bytes of payload - which is a considerably low amount. Packet loss is kept at around less than 10% and, as expected, greater SF values tend to increase air time and decrease the payload size.

And (GHAZALI; TEOH; RAHIMAN, 2021) presents a rich survey about the usage of UAVs in real-time LoRa applications. Not only does it contain many of the technical and theoretical information about its bit rate, spreading factor, transmission power, coding rate, bandwidth and carrier frequencies around the world, a plethora of papers showing UAV-base modules and UAV-based gateways are also evident within the survey. Some of the most interesting are: (DAVE et al., 2020), that focuses on the detection of gas leakages with air and humidity sensors, with a camera feed enhanced via machine learning; (TRASVIÑA-MORENO et al., 2017) in which a wireless sensor network is proposed for marine environmental monitoring, with compelling real-life results; and (ZORBAS; O'FLYNN, 2019), a study on adaptable usage between LoRa and Wi-Fi for high-data management in agricultural applications.

2.3 Drone-array Optimization via Metaheuristics

Many articles are available in the literature about the employment of metaheuristic computation and machine learning in order to coordinate or facilitate the control, pathing and positioning of drone-arrays.

One of such is found in (AL-TURJMAN et al., 2019), whence optimization methods such as the genetic algorithm (GA) and simulated annealing (SA) with the intent of better positioning UAV base stations (UAV-BS) propagating in the 5G spectrum. Results of the simulation are then analyzed for an area of 80 square kilometers, and the objective function to be minimized is the difference between the total area and the integration of areas covered by each of the UAVs.

Whereas in (KALANTARI; YANIKOMEROGLU; YONGACOGLU, 2016), a heuristic algorithm has been proposed to function on the optimization of drone positioning in a 3D environment, in order to supply good area of coverage and data rates for densely-populated user regions. The chosen optimization method in this paper was particle swarm optimization (PSO), and allied with its inherent velocity, results are very promising. The algorithm can also determine whether a number of drones is satisfactory or redundant, as well as the maximum data rate of users which can be served by a UAV.

It seems that many of the works found in the literature for drone-array coordination and positioning are, in fact, based on PSO and its variants. For instance, in (YUHENG; LIYAN; CHUNPENG, 2019), PSO is utilized to perform a simpler task of finding an optimal position in a 3D space for UAVs in emergency network situations. Similar results are attested in (AGGARWAL; GOYAL, 2021), in which UAVs are deployed for disaster management situations. However in (ZHICAI; JIANG; HONG, 2021), a cooperative search algorithm (based on PSO) optimizes the density and distance between micro UAVs in a swarm-based micro drone system. The objective of the study is to prevent UAV clustering in heavily occupied areas, thus distributing some of the drones to underpopulated areas within the search space - potentially avoiding local optima and widening the system's coverage area.

There are, however, studies that utilize other kinds of machine learning for drone optimization. One such example is (RAJASHREE; RAJESH; ANNEPU, 2019), where it exposes another kind of bioinspired technique - the Spider Monkey algorithm - which is quite a rare solution. It is used to compensate for the high localization error made by geolocation via received signal strength (RSSI). This can be a money-saving measure, as installing GPS sensors in each drone or on-ground signal receivers, although being a slightly more precise method, could be much more costly. The paper, however, gives little detail about the Spider Monkey optimization and its creation, with shallow comparison between other bioinspired techniques in its Results section.

Some surveys on the usage of machine learning for UAV positioning and signal propagation, which are interesting to denote, are contained in (MOZAFFARI et al., 2019) and (ZHOU; RAO; WANG, 2020). The former contains a plethora of pertinent information about utilizing UAV arrays in communication systems in general, but also draws attention to many articles that deal with trajectory optimization, cellular network planning and channel modeling via machine learning solutions. The latter, thus, exposes papers that prefer to use a swarm-based learning approach. It also contains studies that analyze drone applications in IoT and that aim to optimize area of coverage of signals, which are both related to the work made in this dissertation.

2.3.1 Studies with UAV Systems in Forested Environments

Even though there are many examples in the literature on UAV usage for sensoring and monitoring forested locations, the same cannot be stated about studies on the higher demand and high-power wireless networks for internet users - which is considerably scarce. This may be due to less necessity (in the past) compared to works that act on urban and suburban environments, having different kinds of diffraction, shadowing and obstructions to line-of-sight of signals. However, in order to make cellular networks available to anyone, anywhere, those kinds of scenarios are now in need of further study and consideration.

And since the work herein has the objective of tuning in UAV arrays for both sensor and internet network usage, papers on both subjects are to be denoted in this subsection.

For high-power, internet-focused applications in forested areas, one has to generally look into non-line-of-sight (NLOS) studies that can corroborate channel and path loss modeling that is compatible with this environment. In a survey by (KHAWAJA et al., 2019), some examples in wooded and forested suburban locations, and even for the modeling of shadowing caused by vegetation are found. The most interesting, for the implementations in this article, are: (CID; ALEJOS; SANCHEZ, 2016), (KHAWAJA; GUVENC; MATOLAK, 2016) and (NEWHALL et al., 2003).

Generally, for sensing and imaging of forested areas, UAVs are widely utilized. Some of works include "remote sensing", where important factors such as vegetation density and forest fires are detected by imaging and outside data obtained via drones. And there are papers that demonstrate on-ground sensoring of forested, rural and green suburban areas, which are more aligned to the main objectives of the dissertation study.

Several remote sensing works in forest environments have been published, but it is enough to briefly point out three samples seen in a review article by (TORRESAN et al., 2016). In (AICARDI et al., 2016), UAV multi-temporal imaging and reconstructed digital surface models throughout 2008 to 2015 have been analyzed to evaluate the post-fire recovery of a secondary forest in northern Italy. Whilst in (MICHEZ et al., 2016), an also multi-temporal analysis is drawn to differentiate the density occupied by different tree species, as well as their health conditions, in a riparian forest inside Belgium. Lastly, an article by (PIERZCHAŁA; TALBOT; ASTRUP, 2014) discourses on the results of post-deforestation and post-harvest soil displacement in a steep, mountainous area of Western Norway; aerial imagery via drones and a 3D reconstruction of the affected terrain was made to estimate soil volume and displacement in this 7-hectare perimeter.

An example of a valuable paper in regards to air-to-ground sensor communications in a rural forested environment is (DUANGSUWAN; MAW, 2021). Within, a comparison on the prediction of path loss in a precision-farming sensor network is made, between PSO and two machine learning algorithms: k-Nearest and Random Forest. The objective functions and equations pertaining to the output of all algorithms is explained in detail, as well as the training processes involved. However, even though the theoretical part supports multiple UAVs, measurements have been made in a small area of 50 x 30 meters, with only one UAV and one IoT sensor. However, the path loss prediction shows satisfactory results, and the Random Forest algorithm performs with the lowest root mean square error (RMSE).

2.4 Final Considerations

In this Section, many works present in the literature are exposed, which have influenced this Master's dissertation. It is also evident that the dissertation draws from several sources and pertains to many areas of interest, such as, mainly, the ones from all three subsections. The major contribution that the work herein aims to achieve is the analysis between simulated and measured signal propagation results in light-to-heavily forested environments via UAVs, using the campus of the Federal University of Pará (or UFPA) as a reference within the Amazon rainforest area. The optimization technique is then applied to position these UAVs in order to better attend the needs of users, and maximize coverage area inside of this context.

3 Low Power Wide Area Networks and LoRa

3.1 Initial Considerations

In this chapter, the theory behind Low Power, Wide Area Networks (LPWAN) and LoRa is to be explained in a succinct manner. Some topics within this subject include: types of LPWAN, wireless sensor networks, and the operation of LoRa.

3.2 LPWAN and Their Types

Low-power, wide-area networks are means of wireless communications that act on low-data requirements, generally for the requirements of power saving and greater signal range whilst sacrificing data rates. With the advancement and necessity of this technology, there are, nowadays, many kinds of LPWAN - and the majority of them are utilized in the context of Internet of Things (IoT) and wireless sensor applications. Many of the papers in the previous Chapter have been displayed on the usage of said networks in order to promote sensoring, evaluation and activation of IoT devices. The greater emphasis has been put on LoRa, which is the technology and frequency spectrum utilized in this work - and that shall be explained in further detail at its own section.

In broad terms, LPWAN are applied in situations which low-power usage and an extensive range is required, which is contrasted to other technologies such as, e.g., Wi-Fi (high-power, low-range), LTE (high-power, high-range) and Bluetooth/NFC (low-power, low-range).

According to (XU; NI, 2021), there is a classification of two major types of LPWAN according to their spectra: ones that do not have a licensed frequency band (LoRa and SigFox) and the ones that operate at licensed spectra, generally supported by 3GPP protocols (NB-IoT and LTE-M). Both types are well utilized in communications and IoT settings, however, many recent studies favor unlicensed LPWANs, as they are easier to implement and research. Therefore, some of the most frequently found LPWAN systems are to be explained below.

SigFox is an ultra-narrow band (UNB) LPWAN designed and implemented by the company of the same name. It is mainly utilized for medical, industrial and research-aid applications, as attested in studies such as (CRYS; HWEE-PINK, 2019) and (RIBEIRO et al., 2018). Since it has such a narrow band of operation, data rates tend to be extremely low, making it suitable to applications which need only a few bytes of data. It may not be suitable for operating an array of drones or a sensor network with dozens of nodes. However, power consumption is kept to a minimal and its coverage area tends to be the greatest compared to all LPWAN, as denoted in (ALDAHDOUH et al., 2019).

As a network within a licensed spectrum, the NB-IoT technology also deals with high-coverage, low-power and narrow-band transmissions. Its operation involves frequencies in various LTE bands and elements from LTE and 3GPP standards are employed as well. Generally, it is designed for IoT applications which require a greater amount of data (around kbps) and connection speed, and that can benefit from the quality of service (QoS) provided by 3GPP protocols, as seen in (MARTINEZ et al., 2019). Yet, its range is lower than LoRaWAN and significantly lower than SigFox, and it consumes more power, since transmission power is slightly higher than those.

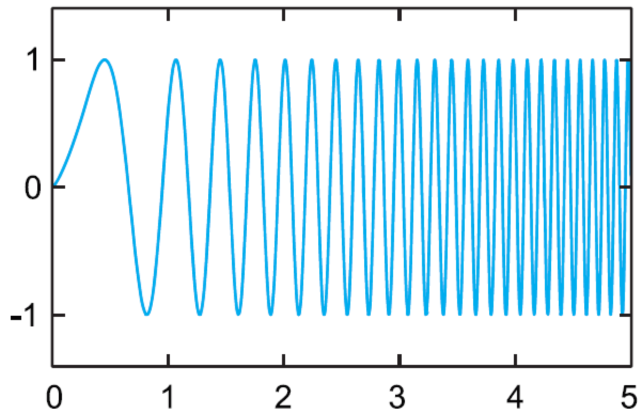
LTE-M (or Long Term Evolution for Machines) is the oldest of the LPWAN models, having backwards compatibility with 4G-LTE technologies and has been developed parallel with NB-IoT and released at the same time. It possesses data-rates akin to 4G systems, mostly reaching around 1 Mbps. Therefore, it supports a good quality of audio/voice and low resolution video, and studies made using these properties of LTE-M have shown to be sucessful, such as the one found in (HENG; HONGWEI, 2019). Also, (BORKAR, 2020) further explains the resources available in LTE-M, its parallels to NB-IoT and its adaptation to fit into 5G standards, which are all interesting discussions.

Having discoursed on the prominent details of LPWAN models, a deeper dive into the functioning and features of LoRa is analyzed in the following section.

3.3 The Operation of LoRa

LoRa, or LoRaWAN, is a standardized type of LPWAN modulation that focuses on the communication of low-data devices that need a greater coverage area than the ones proposed by licensed-band ones. It has been assembled by the company of the same name, located in France. It is also derived from a modulation technique known as Chirp Spread Spectrum (CSS) in which the transmitted signal varies in frequency, inside the allocated frequency spectrum, making it extremely robust to interference. An example of a wave modulated under CSS method is found in Figure 1.

Figura 1: Example of a wave signal modulated by Chirp Spread Spectrum technique.



Source: (SEMTECH, 2019a)

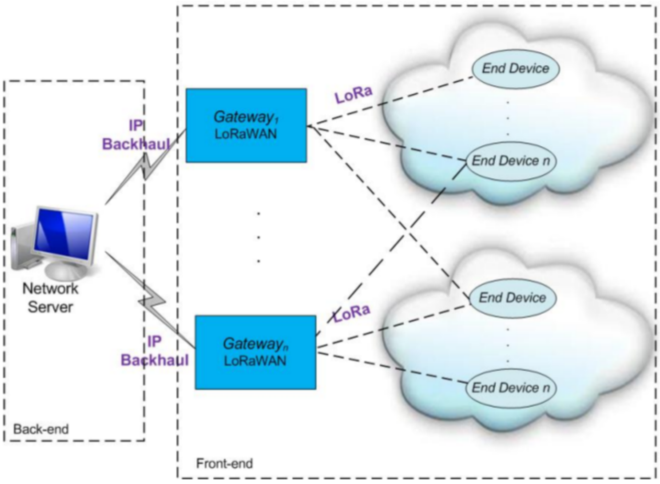
LoRa signal modulation is thoroughly explained in (SEMTECH, 2015). It is written that "In LoRa modulation the spreading of the spectrum is achieved by generating a chirp signal that continuously varies in frequency. [...] The frequency bandwidth of this chirp is equivalent to the spectral bandwidth of the signal". Thus, the equation which denotes the signal's nominal bit rate is:

$$BR = SF.\frac{\frac{4}{4+CR}}{\frac{2^{SF}}{BW}} \tag{3.1}$$

In which BW is the spectral bandwidth, SF is the so called spreading factor of the spread spectrum modulation - which is a constant with discrete values between 7 and 12 - and CR is the code rate that serves as an error-correction constant with discrete values from 1 to 4. So this is the modulation as already corrected in order to improve signal robustness. Given that these are all constant-dependent values, this formula can be transformed into table of reference for convenience. (LAVRIC; POPA, 2017), for instance, has a table that informs not only bit rate, but also the correct SINR limit, time on-air in seconds for 10-byte packets and number of symbols (bytes) per second for every SF value. Please note that these values reflect a bandwidth of 125 KHz.

A basic explanation on the architecture of LoRaWAN is given in (LAVRIC; POPA, 2017). It possesses a star-of-stars network topology. The system has a central network server (back-end), which acts as a hub to connect one or more LoRa gateways that, finally, receives, transmits and manages data from the network users (front-end). Some users may communicate with more than one gateway if necessary. Figure 2 denotes the architectural topology of LoRaWAN.

Figura 2: LoRa topology and architecture



Source: (LAVRIC; POPA, 2017)

LoRaWAN is considered to have great scalability, simply because multiple gateways can be implemented if the network needs to cover more users/devices. Low-cost gateways can support 8 consecutive users, just like the one used to provide measured data for the dissertation herein. This is also taken into account in the simulations, discoursed about in Chapter 5. However, there are also a 16-channel mobile gateway module and a 64-channel gateway intended for installation in cell towers and building rooftops, according to (SEMTECH, 2019a).

3.4 Final Considerations

Brief explanations have been given for the differences of LPWAN, especially LoRaWAN, to other types of systems. Being able to carry low amounts of data for an extensive range, they tend to diverge greatly from Wi-Fi in terms of application, but both can be utilized, in distinct contexts, to employ IoT implementations.

That being said, it is important to denote that SF and other information specific to LoRa will be considered when simulating the drone array system in MATLAB. Chapter 5 ahead shall explain how these theoretical concepts tie up to produce the aforementioned simulation environments for both LoRa and Wi-Fi. Furthermore, it shall demonstrate a path loss model for LoRa propagation that has been calibrated with data taken from the measurement campaigns at UFPA.

Up next in Chapter 4, the theory behind the three bioinspired algorithms utilized in simulations will be discoursed.

4 The Bioinspired Optimization Techniques

4.1 Initial Considerations

Bioinspired computational methods are mainly based on natural selection. They are set to mimic the natural behavior of nature, in which the best and most surviving individuals prevail. With that in mind, these bioinspired algorithms serve as good optimization methods for mathematical and engineering problems, especially those where metaheuristic techniques (trial and error) can be applied to achieve one or more concrete goals.

In this chapter, all three computational optimization methods are to be explained in plain detail, with detailed set of instructions, pseudocodes and their respective equations. In order of appearance these are: the cuckoo search (CS), the flower pollination algorithm (FPA) and the bat echolocation algorithm (BA). These have been chosen as alternatives to more conventional techniques such as genetic algorithms (GA), differential evolution (DE) and particle swarm optimization (PSO), as well as being slightly faster in solving non-linear, multi-variable problems. They have all been designed and implemented by Xin-She Yang (YANG, 2020). The cuckoo search algorithm was the first to be released, in 2009. Then, around 2011, the bat echolocation algorithm has been published, followed by the flower pollination algorithm a year later.

In the context of our study, the variables to be solved are the coordinates of all drones within the array (x-axis, y-axis, height). On certain heavier application, i.e. when a greater number of drones is to be required in simulation, these algorithms must act swiftly and precisely to be able to maximize potentially demanding and numerous drone-array systems.

4.2 Cuckoo Search

The Cuckoo Search algorithm (CS) has been coined by Yang and Deb in 2009 (YANG; DEB, 2009), proving itself to be a very effective metaheuristic algorithm for all sorts of applications in mathematics, industry and engineering (SHEHAB; KHADER; AL-BETAR, 2017). The main bioinspired idea behind this method is the computational modeling of the parasitic behavior of cuckoo-type birds, that often lay their eggs inside the nests of other types of birds. This is a natural occurrence in nature, as the host species often does not perceive the cuckoo's "alien" egg inside the nest, or either choose to ignore it completely or abandon the nest altogether, choosing another place to lay its eggs on. However, if the egg is ignored and left to grow inside the host bird's nest, the cuckoo hatchling is born, and reaches maturity much faster than the other eggs, pushing the host bird's eggs outward. Thus, the cuckoo baby bird expels the other eggs from the nest, resulting in a higher food share for it, and becoming well-fed.

Thus, according to the creators of the CS algorithm (X-S. Yang and S. Deb), the whole process is based upon three major rules:

1. Each cuckoo lays one egg at a time, and deposits it in a random nest. Each egg is considered a potential solution – metaheuristic
2. The best nests carry the best eggs (solutions), and these will survive the next generations due the parasitic nature of the cuckoo hatchlings – elitism.
3. The number of available nests is constant, and defined by the code developer. The probability of a cuckoo egg being discovered by the host bird is defined as $Pa \in [0, 1)$. After this, the bird may choose to discard this egg or abandon its nest – discarding the worst solutions.

The latter rule can also be described by indicating that a probability fraction Pa from the various n nests of host birds are replaced by new nests, presenting randomized solutions.

The cuckoo birds in this method move according to the so-called Lévy Flights (YANG; DEB, 2009). This device provides a random flight path that which each cuckoo (in a number of i cuckoo birds) will trek to find nests. Equations (4.1) and (4.2) are a mathematical representation of the Lévy Flight and Lévy Distribution, respectively, to be implemented in code form:

$$X_i^{t+1} = X_i^t + \alpha \oplus Levy(\beta) \tag{4.1}$$

$$Levy(\beta) \cong u = t^{-(\beta+1)}; (1 < \beta \leq 3) \tag{4.2}$$

In which i is the maximum number of cuckoo birds in the current generation, and t is the current code iteration. Constant α is the step size to be utilized in the code, and is adaptable to the developer's need – it must always be greater than zero, and in this study the value is set to $\alpha = 1$.

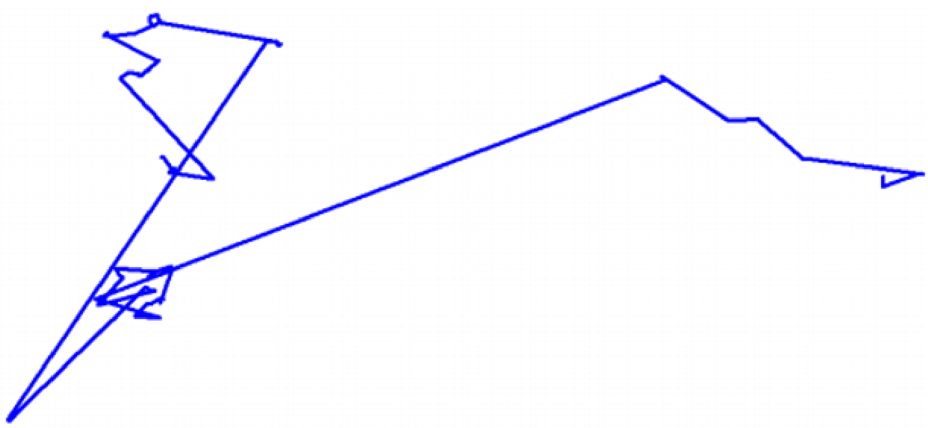
In equation (4.1), the Lévy Distribution is associated to the Lévy Flight with the product $\oplus$, which means "entrywise multiplication" – a kind of product between two matrices of the same size. In the PSO algorithm, a similar product can be found, however, for the Lévy Flight method, the search space can be much better harnessed.

As for equation (4.2), this is about the Lévy Distribution. It possesses infinite variance and average values. The variable β is the random step length, needed for providing a variable magnitude to the random walk performed by the Lévy Flight.

Given that the Lévy distribution presents infinite variance, the search space is virtually limitless, meaning that the length of the flight taken could be very short or incredibly long. However, generally, the new solutions are generated through the Lévy method around the best obtained solution on a given instant, accelerating the process and concentrating the computational effort in a part of the search space. Oftentimes, however, solutions are generated randomly across the space – this is good to prevent the algorithm from getting stuck in local optima, which are not the globally best possible solution.

An illustration showing a Lévy flight with around a hundred steps can be found in 3. The nature of its short and long steps is clearly visible, and its infinite variance may aid to even out and potentially exceed by small margins the search spaces which are to be set in the drone simulations. This can be advantageous, as the work proposed herein present network user sets generated by normal distribution, and may also, in some cases, slightly surpass the optimization bounds. Lévy flights, therefore, could design solutions able to attend the needs of said out-of-bounds users, as it shall be attested in the results of the simulations, in Section 6.

Figura 3: Representation of a 100-step Lévy flight in its search space



Source: (NETO, 2015)

A pseudocode of the CS algorithm, as well as its multiobjective counterpart, can be found in the article in which it was proposed by Yang and Deb (YANG; DEB, 2009). Below, in algorithm 1, a transcription of this code is presented.

Algoritmo 1 Cuckoo Search Algorithm

Define the objective function as $f(x), x = (x_1, ..., X_d)^t$
Generate the initial population of n host nests $x_i(i = 1, 2, ..., n)$

while ($t <$ number of iterations) or (stop criterion) **do**
  Select a cuckoo randomly via Lévy Flight ▷ (4.1)
  Evaluate the cuckoo's fitness (represented by F)
  Draft a random nest (say, j) out of the n available

  if $F_i < F_j$ **then**
    Replace randomly drafted j by the new solution x_i
  end if

  Discard a fraction P_a of worse nests and build new ones
  Keep the best / better quality solutions
  Rank the best nest set and find the current best

end while

Post-process and visualize results

4.2.1 The Flower Pollination Algorithm

A solution based on the behavior of natural flower pollinators has been proposed by Yang (the same author of the cuckoo search and the bat echolocation algorithms), which also utilizes the Lévy flight method of optimal space search. Its efficiency, in many single and multi-objective applications is proven to be greater than Particle Swarm and Genetic optimization algorithms (YANG, 2012).

As for the stages of the algorithm, four rules are defined thusly (YANG, 2012):

1. Biotic and cross-pollination is considered as global pollination process with pollen-carrying pollinators performing Lévy flights.

2. Abiotic and self-pollination are considered as local pollination.

3. Flower constancy can be considered as the reproduction probability and is proportional to the similarity of two flowers involved.

4. Local pollination and global pollination is controlled by a switch probability p $\in$ [0, 1]. Due to the physical proximity and other factors such as wind, local pollination can have a significant fraction p in the overall pollination activities.

Two kinds of pollination are considered and simulated: global and local pollination. This assures that the code does not only fall for local solutions, healthily seeking to encounter a global solution to the objectives. For simplicity manners, the algorithm is based on the idea that every plant possesses only one flower and can pollinate also just one other flower at a time, when in true biological terms they can hold a few flowers and millions of pollinating gametes. This is so that a plant / flower / pollinating gamete are all considered to be part of one solution altogether.

Hence, the first rule (global pollination) and third rule (flower constancy) of FPA are mathematically represented as shown in (4.3):

$$X_i^{t+1} = X_i^t + L(X_i^t - g*) \tag{4.3}$$

Where X_i^t is the pollen i or solution vector X_i at iteration t, and g* is the current best solution among all solutions at the current iteration.

Pollination strength L is dealt via Lévy flight, in which it is a measure of each flight's step size, as denoted in (4.4). Here, the flights symbolize the path of insects and pollinator animals in a given area - in the algorithm, this area is the optimization's global search space. However, the equation used in this algorithm differs from the one found in cuckoo flights, as it is based on producing Lévy flights via the Mantegna algorithm. This is basically a technique to generate pseudo-random step sizes via normal distributions in order to provide an optimal performance whilst still maintaining the demands of the Lévy distribution.

$$L \approx \frac{\lambda\Gamma(\lambda)\sin(\pi\lambda/2)}{\pi s^{1+\lambda}}, \tag{4.4}$$

in which $\Gamma(\lambda)$ is the standard, classic gamma function found in Lévy flights, and other probabilistic and complex number applications.

The Mantegna step size algorithm can be explained as the equation (4.5).

$$s = \frac{\lambda\Gamma(\lambda)\sin(\pi\lambda/2)}{\pi s^{1+\lambda}}, \tag{4.5}$$

with s being the step size, U being drawn from a Gaussian distribution of variance σ^2 and V also being drawn from a Gaussian distribution but with unitary variance, as can be verified in (YANG; KARAMANOGLU; HE, 2014). Generally, the lambda is treated as a parametric value and it is safe to assume that it is a constant with a possible value of around $\lambda \in [0.5, 1.5]$. When $\lambda = 1$, the variance also equals 1, and results are in such case easier to predict.

For the purpose of Rule 2 (local pollination), the flower constancy is mimicked for a limited neighborhood near to the reproductive flower's position. It is represented as

$$X_i^{t+1} = X_i^t + \epsilon(X_j^t - X_k^t), \tag{4.6}$$

where X_j^t and X_k^t are pollens from different flowers of the same plant species.

The fourth rule is a probabilistic switch between global and local pollination, and the probability p can be parametrically and singularly adjusted to improve optimization performance, depending on the need of the objective function.

All stages of the algorithm are represented in pseudocode form by the recommendations in (YANG, 2012), which are transcribed in algorithm 2. Some details previously discussed can be noticed, such as an if/else switch for global and local pollination, which are done by Lévy flights and random selection, respectively.

Algoritmo 2 Flower Pollination Algorithm

```
Define the objective function as f(x), x = (x_1, ..., X_d)^t
Generate an initial population of flowers/pollens as random solutions
Find, within this population, the best solution g_*
Define the switch probability p ∈ [0, 1]

while (t < number of iterations) or (stop criterion) do
    for all n flowers in the population do
        if rand < p then
            Generate a step vector L which obeys a Lévy distribution
            Execute global pollination as in (4.3)
        else
            Pick a uniformly distributed number from ϵ = [0, 1]
            Randomly choose individuals j and k from all solutions
            Do local pollination according to (4.6)
        end if

        Evaluate new solutions
        Update solutions that are better into the population

    end for

    Find the best current solution, represented by g_*

end while
Post-process and visualize results
```

4.3 Bat Echolocation Algorithm

The bat algorithm (BA) is based on the echolocation characteristics of bats, that is, the supersonic frequency of obstacle detection. Basically, the more proximate a bat is from a solution, the algorithm shall "feel"it by a simulated echolocation perspective. Therefore, it uses a search space pathing based on a uniformly distributed echolocation system, not on Lévy Flights.

This algorithm, compared to the other two flight-based one already exposed, has the fastest computation and thus produces results in a more time-efficient manner. This is due to its low calculation complexity, in which individuals do not "evolve", but are simply "swarming"towards a solution. According to (YANG; HE, 2013), there are some parallels to be drawn between the behavior and faster performance of this algorithm and the more classic particle swarm optimization (PSO). Since one of PSO's drawbacks lie in multimodal optimization performance (which is something that applies to multi-drone positioning optimization), the bat echolocation algorithm presents a major improvement in that regard, and this justifies its usage as a replacement, or even an enhancement, compared to PSO.

The following rules apply to the bat algorithm (YANG, 2011):

1. All bats use echolocation to to sense distance, and they also "know"the difference between food/prey and background barriers in a sort of "magical"way;
2. Bats fly randomly with velocity v_i at position x_i with a frequency f_{min}, varying wavelength λ and loudness A_0 to search for prey. They can automatically adjust the wavelength (or frequency) of their emitted pulses and adjust the rate of pulse emission r $\in$ [0, 1], depending on the proximity of their target;
3. Although the loudness can vary in many ways, we assume that the loudness varies from a large (positive) A_0 to a minimum constant value A_{min}.

Bats emit a frequency f to help situate themselves in the environment, they move around in velocity v and possesses a certain loudness A, as well as repeating this process at a rate r (also called pulse emission rate). Equations (4.7), (4.8), (4.9) and (4.10) explain the implications of the process.

$$f_i = f_{min} + (f_{max} - f_{min})\beta \tag{4.7}$$

$$v_i^{t+1} = v_i^t + (X_i^t - X*)f_i \tag{4.8}$$

$$X_i^{t+1} = x_i^t + v_i^t \tag{4.9}$$

$$X_{new} = X_{old} + \epsilon A^t \tag{4.10}$$

In case of (4.7), β is a random vector of normal distribution with values from 0 to 1. For (4.10), ϵ is also a random vector but ranging from $[-1, 1]$.

Finally, in (4.11), the loudness and pulse emission rates are categorized, in which two constants $0 < \alpha, \gamma < 1$ are utilized for further parametric optimization. Please notice that r_i refers to the pulse emission rate, as stated in Rule 2.

$$A_i^{t+1} = \alpha A_i^t; r_i^t = r_i^0[1 - \exp(-\gamma t)] \tag{4.11}$$

In basic terms, the only terms to be manually configured into the algorithm application is the minimum and maximum frequency, the loudness and the pulse emission rate, as all rules stem from those four variables. A general pseudocode of the BA can be found at (YANG, 2011), and is exposed in algorithm 3.

Algoritmo 3 Bat Echolocation Algorithm

```
Define the objective function as f(x), x = (x_1, ..., X_d)^t
Generate the initial population with x_i (i = 1, 2, ..., n) and v_i
Define pulse frequencies f_i for all bats
Initialize pulse rates r_i and echo loudness as A_i

while (t < number of iterations) or (stop criterion) do
    Generate new solutions by adjusting to new frequencies        ▷ (4.7)
    Calculate velocities and the new position of each solution    ▷ (4.8) and (4.9)

    if rand < r_i then
        Pick a solution among the best solutions
        Generate a local solution in the vicinity of the selected best solution
    end if

    Generate a new solution by random flight

    if rand < A_i and f(x_i) < f(x*) then
        Accept the newfound solution
        Increase the pulse rate r_i and decrease loudness A_i
    end if

    Rank the bats in the solution set and find the current best x*

end while
```

4.4 Final Considerations

In this chapter, a detailed explanation into the theory of the chosen optimization techniques has been realized. Furthermore, an explanation of the objective function model, as well as its manner of associating and assigning users for each drone, has been described. In the next chapter, the methodology and application of said algorithms and their common objective function is to be analyzed, further explaining how the study herein has been conducted and laying the foundation to elucidate and validate its results.

For more information on the vast field of research that is bioinspired computing, there is an extensive overview on the subject compiled in a survey by (SER et al., 2019).

5 Project Methodology

5.1 Initial Considerations

The methodology for the project of the simulations and the implementation of the optimization codes is to be discussed in this section. Among many topics, examples are the software and pseudocodes used, explanations about the chosen structure and variables of the simulations, and information on measurement campaigns that produced the empirical propagation model for LoRa.

The techniques and equations that conduct this study must act upon the objectives as set in Section 1.0.2 and 1.0.3. So, mathematical relations must be input into the computational optimization, which shall be distinct for Wi-Fi and LoRa applications, in order to reach these goals. For instance, LoRa optimization and propagation differs from Wi-Fi mainly on its Chirp Spread Spectrum feature, and the implementation of SF values means it may produce results that are not proportional to the basic drone model in Wi-Fi.

Since the methodology of this work is an extensive subject, this part is divided in many subsections, which are summarized below.

Subsection 5.2 discourses about the path loss and propagation modeling utilized as foundation to the results of this work. Subsections 5.2.1, 5.2.2 and 5.2.3 represent, respectively, the theoretical Wi-Fi propagation model, the theoretical LoRa model, and the LoRa model corrected by measured data. Furthermore, 5.2.3 explains the conduction of the measurement campaigns within the forested UFPA campus, and in further detail provides explication on the transformation of said measured results into a log-distance propagation model formula that complements the theoretical UAV-guided LoRa model.

Subsection 5.3 clarify the structure of the algorithms which are used as the search space optimizers and that provide the estimate drone positioning. The theoretical functioning of the bioinspired optimizers is already laid out in Section 4, so subsections 5.3.1 through 5.3.3 focus on the specific structures added upon the algorithms for drone-array optimization for, respectively, the Wi-Fi theoretical model, the LoRa theoretical model and the measurement-corrected LoRa model. This will mainly lay emphasis on how each of said applications provide a different user-drone association function, that influence directly into how and which drones will connect to a group of users. Since the three computational techniques are generally similar in terms of structure, assigning one pseudocode for each of the three applications is enough.

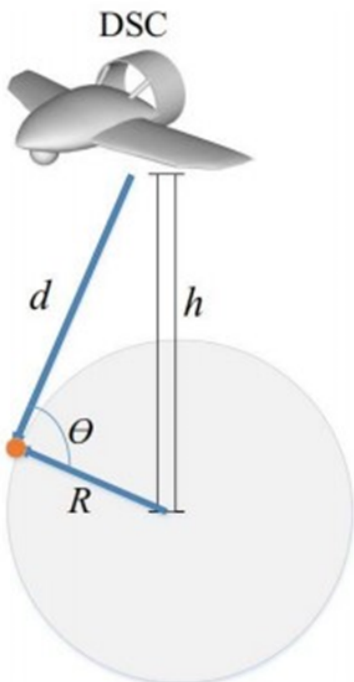
And finally, Subsection 5.4 provide the closing remarks of this section.

5.2 Path Loss and Propagation Modeling of Drones

Herein lie the mathematical equations of propagation modeling of wireless signals utilized in the algorithms. A strong theoretical basis for these is of great importance, as it signifies, in the optimization process, the capacity of the algorithms to perceive a user as connected or not, and to which drone should it connect to, based on the measurements of received power (P_r, as is most referred to in the text) and signal-to-interference-plus-noise ratio (SINR).

The trigonometric equations on the positioning of drones, valid for all applications, are found below in (5.1) and (5.2). Figure 4 is used to represent the trigonometric variables visually. These equations are found in the work of (MOZAFFARI et al., 2015), which is, overall, one of the most utilized propagation models for UAV systems in the literature, as well as extensively within the author's several articles.

Figura 4: Drone positioning variables



Source: (FERREIRA et al., 2021), (MOZAFFARI et al., 2015)

$$d = \sqrt{R^2 + h^2} \tag{5.1}$$

$$\theta = \arctan \frac{h}{R} \tag{5.2}$$

In which R is the distance from the projection of the drone on the user plane to the user itself, h is the UAV height in relation to the user plane, d is the actual distance from user to UAV and θ is the elevation angle of the UAV in relation to the user.

5.2.1 Wi-Fi Theoretical Propagation Model

The drone modeling for Wi-Fi frequency propagation is done in this study according to (MOZAFFARI et al., 2015), in which a path loss model for UAV-specific operation is described. The model conveys both Line-of-Sight (LoS) and Non-line-of-sight (NLoS) losses and is represented in equations (5.3) and (5.4), and probability of having a LoS connection for an elevation angle of θ is given by (5.5).

$$L_{LoS(dB)} = 20 \log \left(\frac{4\pi f d}{c}\right) + \zeta_{LoS}, \tag{5.3}$$

$$L_{NLoS(dB)} = 20 \log \left(\frac{4\pi f d}{c}\right) + \zeta_{NLoS}, \tag{5.4}$$

$$P_{LOS} = \frac{1}{Z}, Z = 1 + \alpha \exp(-\beta[\frac{180}{\pi}\theta] - \alpha) \tag{5.5}$$

In which f is the propagated frequency and ζ_{LoS} and ζ_{NLoS} are loss constants related to LoS and NLoS propagations. Furthermore, α, β are environmental constants which are necessary to adapt this model to urban, suburban or rural ambiences, and $P_{NLOS} = 1 - P_{LoS}$. Values of d and θ are thus according to (5.1) and (5.2).

Hence, the average path loss between LoS and NLoS situations is described as:

$$L_{avg(R,h)} = P_{LoS} \times L_{LoS} + P_{NLoS} \times L_{NLoS}, \tag{5.6}$$

which is then, by applying eqs. (5.3), (5.4), (5.5) and (5.6), transformed into:

$$L_{avg} = 20 \log \left(\frac{4\pi f d}{c}\right) + \frac{(\zeta_{LoS} + Z\zeta_{NLoS})}{(1+Z)} \tag{5.7}$$

So, considering transmitting and receiving antenna gain, (5.8) denotes the received power of the user based on the path loss:

$$P_{r(dB)} = P_t + G_t + G_r - L_{avg} \tag{5.8}$$

It is important to denote this is a value unique for each user connected to each drone. In light of this, received power between drones in an array system might cause UAV-to-UAV interference and, as such, has to be considered into the model. In that manner, the SINR metric comes into activity, which determines how many users are connected to the UAV array and the quality of its signal. As said previously, SINR is crucial for the algorithm to decide which UAV should connect to which user, and its implementation is given by:

$$SINR_{i,j} = \frac{Pr_{i,j}}{q + \sum_{k=1,k\neq j}^{N_{UAV}} Pr_{i,j}}, \tag{5.9}$$

where received power values for the j-th drone and the i-th user are then transformed into SINR by dividing it for the noise floor level q added to the average of the sum of the received powers of the other drones (that is, the UAV-to-UAV interference). The formula was derived in accordance with SINR information in (GOLDSMITH, 2005) and (CARDOSO, 2015). To calculate this formula more easily, SINR and Pr values should be expressed linearly.

Values for the noise floor can be treated, for instance, as defined by the absolute noise floor formula, given in (5.10) for ambient temperatures (around 300 K, or 27 °C), as seen in (SEMTECH, 2015).

$$q_{(BW)} = -174 + 10\log\left(BW\right) \tag{5.10}$$

Given that bandwidth values of regular Wi-Fi systems are 20 MHz or 40 MHz according to IEEE 802.11 standards (KHANDURI; RATTAN, 2013), values are around $q_{(20MHz)} = -\ 101$ dBm and $q_{(40MHz)} = -\ 98$ dBm.

5.2.2 Theoretical Propagation Model for LoRa

For the theoretical modeling of LoRa, the equations in the Wi-Fi model, (5.3) through (5.10), are to be reused. However, for the signal propagation of LoRa, there is a change in the limits of signal propagation according to its spreading factor values, as a result of the chirp spread spectrum technique. In theory, higher SF values provide greater coverage area and robustness to noise but lower bit rates, and lower values enhance the bit rate of the signal but sacrifices some area of operation.

But the way the SF enhances the coverage area is by lowering the sensitivity needed so that the signal can be received properly. This results in lower RSSI and SINR values being possible for higher SF values, and vice-versa. Thus, it is safe to conclude that SF values should not actually be counted and adapted upon the propagation model itself – it is only necessary to adjust the optimization limits in regards to received power and SINR according to the SF of the transmitted signal. Then, it is only up to the optimizers to encounter optimal values in these ranges. In practice, even though it should not cause a great difference in the results, altering SF values may cause distinct results and alter the propagation modeling, and this is addressed by the corrected empirical model in section 5.2.3.

Table 1 refers to the signal sensitivity range and its respective SF values for a bandwidth of 125 KHz and carrier frequency of 915 MHz, which is the recommended LoRa operation values for Europe and Brazil. The FSK mode (short for frequency shift-keying) specified here is another type of modulation that is also present in LoRa gateways as an alternative to CSS, albeit generally less effective, and it is only listed as means of comparison.

Tabela 1: SF reference values of LoRa for f = 915 MHz, BW = 125 kHz

Mode	**Bit Rate (kbps)**	**Sensitivity (dBm)**	**Δ (dB)**	**Estimated Range**
FSK	1.2	-122	–	N/A
SF 12	0.293	-137	+15	10-12 km
SF 11	0.537	-134.5	+12.5	10 km
SF 10	0.976	-132	+10	8 km
SF 9	1.757	-129	+7	6 km
SF 8	3.125	-126	+4	4 km
SF 7	5.468	-123	+1	2 km
SF 6	9.375	-118	-3	N/A

Source: (SEMTECH, 2015), (SEMTECH, 2019a)

For the simulations, the SF 10 mode has been chosen, as it possesses good concordance between data rates and estimated range to be utilized in its normally-distributed search space environment. More details on Chapter 6. Also, since the bandwidth of LoRa applications for this study is 125 kHz, (5.10) gives a value of around $q_{(125kHz)} = -123$ dBm.

5.2.3 The Corrected Empirical Model for LoRa

Measured data to better represent medium to densely-forested environments has been acquired at Federal University of Pará (UFPA) in a series of measurement campaigns conducted by the staff of LCT-UFPA (in portuguese: Laboratório de Computação e Telecomunicações). The objective was to create a database of LoRa empirical results to be shared by students who work on the subject of drones and LoRa systems, and to utilize these data in the different scientific studies present throughout the laboratory. The study on this dissertation, thus, has benefited from this project.

An 8-user LoRa gateway module configured by an Arduino UNO has been used, as well as two compact omnidirectional antennas with a maximum transmitting power sensitivity of around 10dBm and one drone, to which we attached the gateway, to realize measurements at different heights.

The drone was kept at a fixed place, thus emulating its usage as a LPWAN gateway transmitting data (Tx), as it transmitted down to the receiver antenna (Rx) which varied its position. The Rx was attached to a car, and thus further distancing away from the UAV in order to measure the variation of RSSI and SINR values according to distance and to the LoRa mode in which the gateway was configured.

Figure 5 displays the path taken by the car for all measurements campaigns. The journey encompassed a travelling distance of around 2.9 km, sprawling from UFPA's bus garage (in red) all the way to CEAMAZON (in green). The path chosen is painted in blue, and the position of the drone, fixed in all campaigns, is represented as a purple circle in the picture. Often, measurements would be halted exactly halfway in order to exchange the drone's battery packs, and then resume normally. A photo showing the Rx antenna attached to the car and the utilized drone in mid-air is represented in 6. Notice that there is a blue 3D-printed module on the back of the drone, made to attach it to the LoRa gateway.

SF values of 8, 9, 10 and 11 were selected, as well as heights of 6 meters, 24 meters, 42 meters and 60 meters above ground. This amounts for a total of 16 measurement journeys, and a total of 2824 points measured.

For the estimation of a path loss model for the measured data obtained, the log-distance path loss model has been used (abbreviated to LDPL, in this dissertation). In the work in (CHALL; LAHOUD; HELOU, 2019), they have used some of the different

Figura 5: Path taken in all measurement campaigns, at UFPA

Source: Google Earth, Author

Figura 6: Setup showing the receiving antenna, and drone with LoRa gateway

Source: Author

path loss models to compare which would best fit their measured results - the log-distance one being a part of them.

The equations, thus, have been derived from both measured data and the classic LDPL model. Its general equations go as follows:

$$L_{LPDL} = L_0 + 10nlog(\frac{d}{d_0}) + X, \tag{5.11}$$

$$Pr_{LPDL} = P_t + G_t + G_r - L_{LPDL}, \tag{5.12}$$

in which d_0 is the reference distance,L_0 is the path loss in the reference distance, n is the path loss exponent (PLE), d is the distance or length of the path, Pr_{LPDL} is an estimate of the received power with zero antenna gains and X is a normal random variable with mean equaling to zero, which is supposed to emulate the fading effects of signal loss.

From (GOLDSMITH, 2005), values for signal amplitude and, therefore, inference of the proper path loss model used to approximate modeled results to measured ones are then transcribed into (5.13) and (5.14):

$$A = 10log\frac{d}{do}, \tag{5.13}$$

$$n = A\backslash(-RSSI_{(measured)} + L_0), \tag{5.14}$$

in which A is a distance to reference distance ratio for log-distance and free space models, that is to be inputted into the path loss equation. Given that these calculations need to be input in vector or matrix form in MATLAB, there is in (5.14) a matrix left division symbol, which is necessary to yield the correct results. Furthermore, $RSSI_{(measured)}$ is the vector of received power values, or RSSI, featured in the measured data. Hence, the path loss exponent is estimated.

Next, for the calculation of the standard deviation σ to be utilized in random variable X, the mean square error between measured results and calculated ones must be employed. So, the equation goes as the following:

$$MSE = \frac{\sum_{n=1}^{N_{data}} (RSSI_{(measured)} - p)^2}{N_{data}};, \sigma = \sqrt{MSE} \tag{5.15}$$

where p is the calculated result of RSSI as in (5.12) but without the random variable included. And since MSE is a variance that is considered to be unbiased, it is enough to take its square root to discover the standard deviation.

Since there are different RSSI and distance values than expected in theoretical calculation, in every set of SF values, path loss exponents may differ for every SF mode. Thus, both figures 7 and 8 represent the log-distance model proposed. As for figure 7, it represents the model for only values in SF 10, as it is the value chosen to be utilized in the simulation, whereas 8 displays a calculation over measured data of all SF values.

In Table 2, values of path loss exponent, standard deviation, reference loss and distance found for each SF are discoursed. Reference values have been found by looking for the minimum RSSI value between measured data, for each SF, and annotating its distance. For all SF, reference values are an average of the reference values between all SF. Notice that the high values of PLE of around 4 match empirical observations for outdoor environments with many NLoS losses, as is the case for densely forested areas and not for suburban ambiances, as the model suggested by (MOZAFFARI et al., 2015) utilizes.

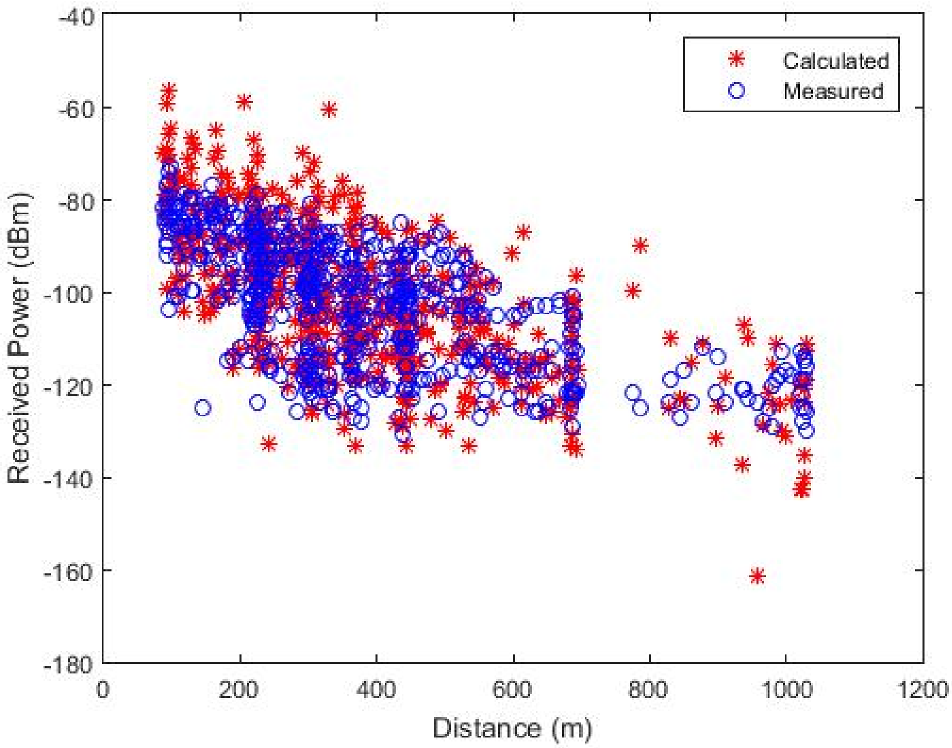
Tabela 2: Values for the LoRa-LDPL model

Mode	**PLE**	σ	**Reference Distance (m)**	**Reference Loss (dB)**
SF 11	4.3	11.75	95.04	77
SF 10	4.63	11.82	94.06	76
SF 9	3.88	13.27	91.51	80
SF 8	4.77	11.26	119.57	75
All SF	4.37	11.8	95.04	77

Source: Author

The curve fitting with values of Table 2 produces, therefore, satisfactory results

Figura 7: Log-distance model for SF 10, expressed in RSSI



Source: Author

that follow the slope of measured data, but with a visible flaw: values for greater distances in measured data all around seem to have smaller variance to calculated results. This opens up possibilities for the enhancement of this model in the future, in order to create more accurate modeling.

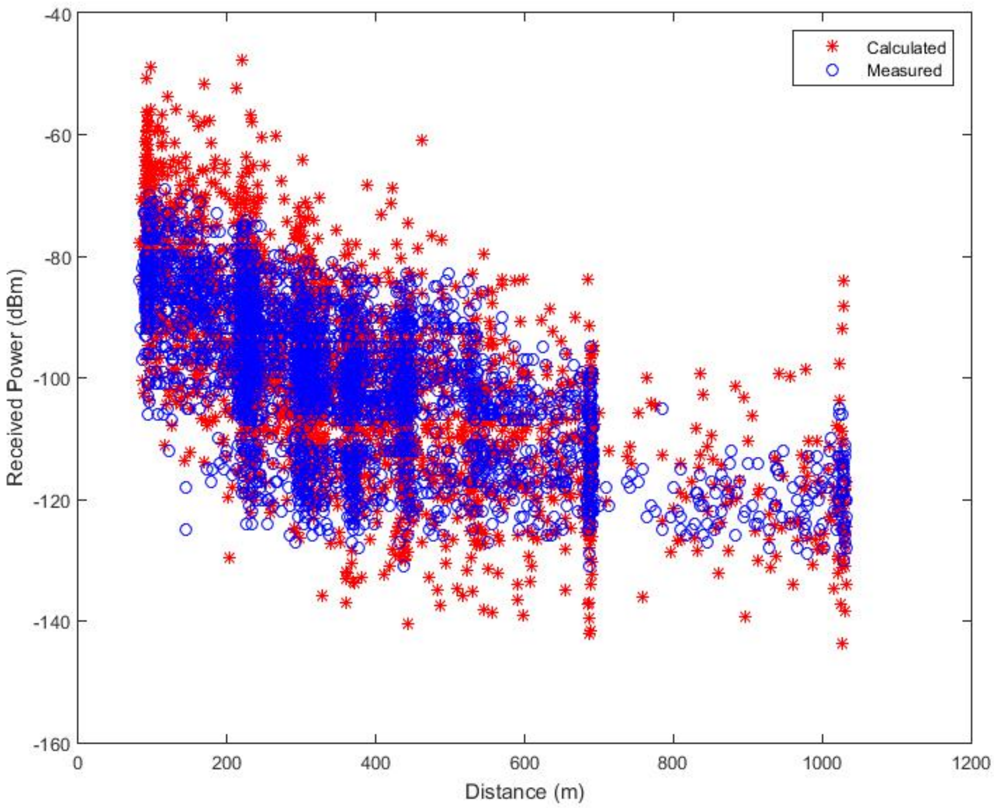
Lastly, since the SINR is an equation that is independent of any path loss model, (5.9) is still applied in this case.

5.3 Structure of the Algorithms

In this section, it is discussed the specific implementations of the three algorithms for each of the three applications. Given that each of the three distinct simulations (Wi-Fi, LoRa, empirical LoRa) possesses different details, variables and constants, it is important to separate the codes into their specific parts. As said previously, the same association function is employed for all three bioinspired algorithms. Thus, only one pseudocode per application (Wi-Fi, LoRa, empirical LoRa) is enough to represent the process - it is enough in the pseudocodes to add a line that represents the choice of the bioinspired technique by the code's user.

There is but one simple, single-variable objective to be optimized in all applications

Figura 8: Log-distance model for all SF, expressed in RSSI



Source: Author

and algorithm implementations, which is the coverage area of drones. Given that the measurements of SINR are fundamental to the signal quality of connection between base-station and user - or gateway and sensors, as is the case for LoRaWAN - it is safe to assume as a standard for all objective functions in this study that the algorithms must always strive for SINR > 0. That is, a drone-array in the optimizations will only recognize a user is connected to it if the corresponding SINR measurement, for any of the drones within the array, is greater than the noise floor and the UAV-to-UAV interference.

Therefore, the number of users connected to the UAV array ($N_{connected}$) is bound by the user's SINR values through a conditional statement. The algorithm of user association via SINR, represented in 4, is to be appended to all simulations.

Algoritmo 4 SINR User Association

for Number of users in the optimization's boundaries **do**
    Obtain SINR values from all drones for the iterated user
    Compare which drone provides the best SINR
    Take and store maximum found SINR value as $SINR_{max}$
    if $SINR_{max} > 0$ **then**
        Recognize user as connected to the drone-array
        Add one more user to number of connected users: $N_{connected}$
    end if
end for

In light of this, the objective function is given in (5.16) as the difference between the number of total users and the number of connected users in the system. Therefore the objective function represents the number of disconnected users in the optimization, which ideally must be approximated to zero.

$$Z_{(SINR)} = N_{users} - N_{connected(SINR)}, \tag{5.16}$$

But in order to obtain the values for this function, we must compute the raw input variables of the positioning of drones until we can obtain SINR values through them.

The input variable quantity is the number of drones in the simulation multiplied by three. Each population iteration will present solutions for drone position data, just as in (5.17):

$$Pop_{vector} = [x_{UAV1}, y_{UAV1}, h_{UAV1}, x_{UAV2}, y_{UAV2}, h_{UAV2}, ..., x_{UAVn}, y_{UAVn}, h_{UAVn}], \tag{5.17}$$

where $[x, y, h]$ are values of the each drone in the simulated environment space. That is, the greater amount of UAVs in the array there are, also greater is the number of inputs - which increases computational cost and time.

Some general equations applied to all algorithms are adaptations to the trigonometrical properties d (distance from user to drone) and θ (elevation angle between user and drone). In order to intertwine the propagation models with the optimization algorithms, values of every population vector are then used to calculate adaptive values of R, d and θ. This relation is given by equations (5.18), (5.19) and (5.20):

$$R_{(i,j)} = sqrt(x_{UAVj} - x_i)^2 + (y_{UAVj} - y_i)^2, \tag{5.18}$$

$$\theta_{(i,j)} = \tanh \frac{h_{UAVj}}{R_{i,j}}, \tag{5.19}$$

$$d_{(i,j)} = sqrtR^2_{i,j} + h^2_{UAVj}, \tag{5.20}$$

which all imply that for all j-th drones and i-th users, the difference of their positioning is what creates the variables of distance in the xy-plane (R), total tridimensional distance (d) and the elevation angle (θ). So, inside the simulation, an (i by j) matrix of these variables is created in relation to every user (index i) and and every drone (index j).

Therefore, by applying these (i by j) matrices in the calculation of path loss, received power and SINR, one can reach the conditional statement elucidated in algorithm 4.

5.3.1 Algorithm for Wi-Fi Theoretical Model

The pseudocode of algorithm 5 takes the association function of algorithm 4 and ties it to the calculation of the Wi-Fi theoretical model explained in Section 5.2.1.

Algoritmo 5 Wi-Fi specific UAV-to-User Association

Define the objective function as defined in (5.16)
Load the normally-distributed set of Users
Define the number of Users and space search bounds
Define number of UAVs N_{UAV} as integer: $N_{UAV} = \frac{N_{users}}{50}$

Execute one of the three bioinspired optimizations (CS, FPA, BA)

Begin association function:

Set constant values necessary to calculate equations (5.1) through (5.10) for Wi-Fi frequency

for (i = 1:1:N_{users}) and j = (1:1:N_UAV) **do**
  Calculate d and θ as specified in (5.20), (5.19)
  Use d and θ to calculate path loss and received power
  Take the matrix of received powers for all drones and users and calculate SINR
end for

Associate users to drone-array via SINR, as seen in 4

Calculate objective function and iterate the bioinspired algorithm

Post-processing and visualization of results (SINR, Fitness)

5.3.2 Algorithm for Theoretical LoRa Model

The are not many issues that differ this implementation of the user-association and bioinspired optimization from the one found in the Wi-Fi application. Main differences attain to the loading constants and variables that are specific to LoRaWAN instead (e.g. LoRa carrier frequency, bandwidth and noise floor), as well as the number of drones utilized.

Since LoRa modules can only work with 8 simultaneous connections, a total cap of 8 users per drone must be fulfilled. So, the pseudocode for the theoretical LoRa simulation is discoursed in algorithm 6

Algoritmo 6 Theoretical LoRa specific UAV-to-User Association

Define the objective function as defined in (5.16)
Load the normally-distributed set of Users
Define the number of Users and space search bounds
Define number of UAVs N_{UAV} as integer: $N_{UAV} = \frac{N_{users}}{8}$

Execute one of the three bioinspired optimizations (CS, FPA, BA)

Begin association function:

Set constant values necessary to calculate equations (5.1) through (5.10) for LoRa frequency

for (i = 1:1:N_{users}) and j = (1:1:N_{UAV}) **do**
    Calculate d and θ as specified in (5.20), (5.19)
    Use d and θ to calculate values of (5.8) and (5.6)
    Take the matrix of received powers from all drones and users and calculate SINR
end for

Associate users to drone-array via SINR, as seen in 4

Calculate objective function and iterate the bioinspired algorithm

Post-processing and visualization of results (SINR, Fitness)

5.3.3 Algorithm for the Empirical LoRa Model

Since there are adjustments to propagation characteristics in the empirically-corrected LDPL for LoRa, the algorithm shall contain the equations related to this change. Values of distance continue to be defined just as in (5.1), as they are not model-specific. But the other equations must change in order to reflect the implementation of the new model.

Therefore, the pseudocode for this implementation of the simulation goes as follows:

Algoritmo 7 Empirical LoRa's specific UAV-to-User Association

Define the objective function as defined in (5.16)
Load the normally-distributed set of Users
Define the number of Users and space search bounds
Define number of UAVs N_{UAV} as integer: $N_{UAV} = \frac{N_{users}}{8}$

Execute one of the three bioinspired optimizations (CS, FPA, BA)

Begin association function:

Set constant values necessary to calculate equations (5.11) and (5.12)

Load σ, reference and PLE values for chosen SF

for (i = 1:1:N_{users}) and j = (1:1:N_{UAV}) **do**
    Calculate d as specified in (5.20)
    Use d, σ and PLE to calculate loss and received power
    Take the matrix of received powers from all drones and users and calculate SINR
end for

Associate users to drone-array via SINR, as seen in 4

Calculate objective function and iterate the bioinspired algorithm

Post-processing and visualization of results (SINR, Fitness)

5.4 Final Considerations

In the methodology chapter, three main focus were considered:

- Explain the model proposed by (MOZAFFARI et al., 2015) which is to be used in theoretical Wi-Fi and LoRa simulations;
- Define the specific values found by a log-distance path loss model proposed by the author, which relates measured data to a propagation model to be used in an improved, empirically-corrected model for LoRa applications in forested environments;
- Lastly, elucidate how the optimization algorithms are structured and coded for the simulations to occur.

The models have been expressed through numerous and thoroughly explained equations. The theoretical drone modeling proposed by (MOZAFFARI et al., 2015) is robust and substantiated by many of his works. So, good accuracy is expected for Wi-Fi simulations which use this model in this study, since his examples of applications follow carrier frequencies very proximate to 2.4 GHz Wi-Fi. But his equations have yet to prove themselves as the best in terms of lower frequency, sub-GHz LoRaWAN systems, and the simulations herein are a way to test this.

Since the log-distance path loss model has been fitted into actual measured data of LoRaWAN propagation, it should be expected to see this model provide best results - even if by small margin.

Also, the simulation environments built in MATLAB provide sound coding for the implementation of the bioinspired techniques and their adaptation to fit calculation of the propagation models that they employ. The process to generate the objective function is shown with caring detail in this Chapter.

In Chapter 6 ahead, results are shown for the simulations made and some comparative statements between applications and bioinspired techniques are to be drawn.

6 Results

6.1 Initial Considerations

In this chapter, results of the simulations discoursed through this entire dissertation shall be exposed. A brief explanation of the objective function settings are to be discussed in Section 6.2, as well as some relevant, general information on the simulation environment built in MATLAB.

There are, as previously informed in other chapters, three sets of simulation values to be analyzed here. Each of them sprawl a number of users by normal distribution in a user plane, and a number of drones defined automatically by the algorithm shall work to maximize the area of coverage via their positioning, in order to attend to many users as possible. The three cases to be analyzed here are: Wi-Fi system with theoretical drone path loss formula, LoRaWAN system with theoretical drone path loss formula and a LoRaWAN system with a measurement-corrected path loss formula.

All three of these cases shall be simulated by the three bioinspired algorithms present in Chapter 4, that in return will execute three distinct sets of generated user data. This is to evaluate both the strength of the bioinspired techniques and the flexibility of the algorithm, as different sets of users require different UAV-array positioning and total number of drones. This amounts to, therefore, 27 simulations.

So, in reference to Section 5.3, the objective function for all algorithms can be defined as:

$$Z_{(SINR)} = N_{users} - N_{connected(SINR)}, \tag{6.1}$$

where the number of users present in the simulation in total is to be minimized by the amount of users connected to the drone array. So, the closer to zero, the better the positioning of UAVs will be. It is characterized as a single-objective system, that is, it only presents one output variable.

In terms of statistical measuring, the relative error C of the objective function can be calculated by the average of disconnected users in relation to the total of users:

$$C = \frac{Z_{(SINR)}}{N_{users}} \tag{6.2}$$

Inputs for the positioning of drones behave just as explained in equations (5.20) and (5.19), in Section 5.3. They are obligatory for the calculation of distance and elevation angle, which in turn are used to obtain the received power and SINR. Since the output depends on the SINR value, the objective function can be simplified to display itself like in (6.1) with just one input.

It is worth noticing that all simulations consider only the drones within the array as mobile. Users are kept to fixed locations, since the number of users is considerably high and would imply in increased complexity for both simulational coding and results analysis.

Also, all simulations have been held in MATLAB, on a computer with 16 GB RAM and an AMD Ryzen© 3.6GHz 4-core CPU. Through trial and error to obtain best results whilst still maintaining short computational time, 200 iterations and 25 solutions per iteration has been selected for Wi-Fi and 500 iteration with 25 solution per iteration has been chosen for both LoRa applications. Locking optimizers in a constant iteration cap makes drawing comparisons between them easier. If the relative error between two optimizers is equal, the best one is chosen by the lowest run time.

Compared to (FERREIRA et al., 2021), where SINR was not taken into account for the drone simulation, its addition has significant weight on computational cost, but also on the accuracy and production of more real-life results. UAV-to-UAV interference is generally very considerable, even so in situations where received signal strength is proximate to noise levels. This dissertation not only takes SINR into account but utilize it as the input variable in optimization.

As for the structure of the chapter, Section 6.2 shall deal with the Wi-Fi application, its drone positioning results for a varying set of number of users and drones in the system; Section 6.3 displays the results of the LoRa system, in which the application is capped at 8 users per gateway and the number of drones vary under that constraint; finally, Section 6.4 denotes results for the formulas found at the empirical LoRa model in Section 5.2.3 - variation of users in simulation is kept to the same number of users as the simulation for the theoretical model, for comparison purposes. Section 6.6 present the author's final remarks for the chapter.

6.2 Theoretical Wi-Fi Simulation

Simulations on a Wi-Fi setting have been conducted on three different sets of users, all of them generated by normal distribution MATLAB function randn and then truncated to the search space bounds. So, the algorithms attempt to optimize sets of 150, 200 and 300 users. Given that many Wi-Fi gateways do not possess a maximum user capacity, it is assumed a number of 50 users per drone.

The lower and upper bounds of the search space have been set as a plane of 16 km^2 (or a 4x4 km surface) centered at (x, y) = (0, 0), as well as for drone heights between 100 and 400 meters.

Below, in Table 3, a list of relevant variables and constants that define the optimizations for Wi-Fi simulation are found.

Tabela 3: Variables and Constants of the Optimizations - Wi-Fi

Parameters	**Values**
Lower bounds (x, y and h) (m)	[-2000; -2000; 100]
Upper bounds (x, y and h) (m)	[2000; 2000; 400]
Number of iterations	200
Solutions per iterations	25
N_{users}	[150; 200; 300]
N_{UAV}	[3; 4; 6]
Switch Probability p (FPA)	0.3
Transmitted Frequency	2.4 GHz
Bandwidth	20 MHz
α	9.6
β	0.28
ζ_{LOS}	1 dB
ζ_{NLOS}	20 dB
N (20 MHz bandwidth)	-101 dBm
G_t and G_r	0 dB
Transmitted Power (all UAVs)	20 dBm

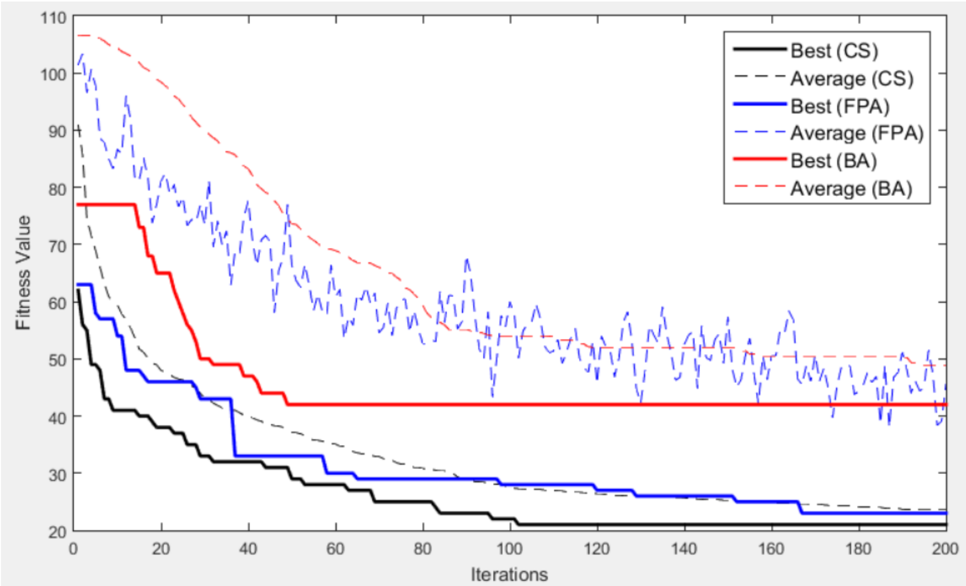
Source: Author

It can be seen that transmitted power has been kept at 20 dBm, which is a common full-power transmitting value for Wi-Fi wireless modems and antennas. α and β are chosen to reflect suburban, lightly-forested areas, and are the most readily found constant values, so the simulation is run on them.

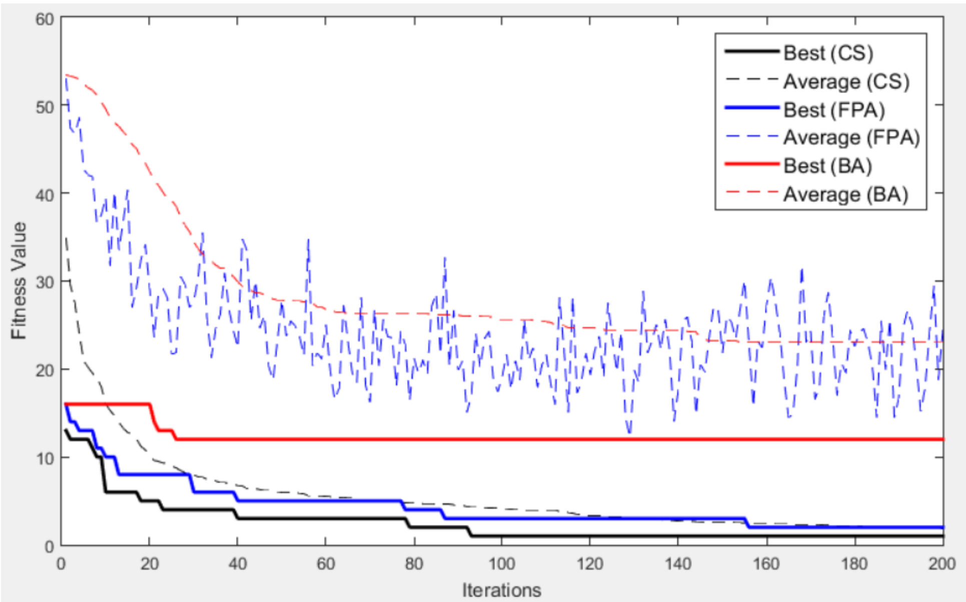
6.2.1 Wi-Fi Fitness and Propagation Results

The fitness evaluation of all algorithms, divided by user sets are present in Figure 9. Notice that better fitness results are expressed when there are more drones covering the area.

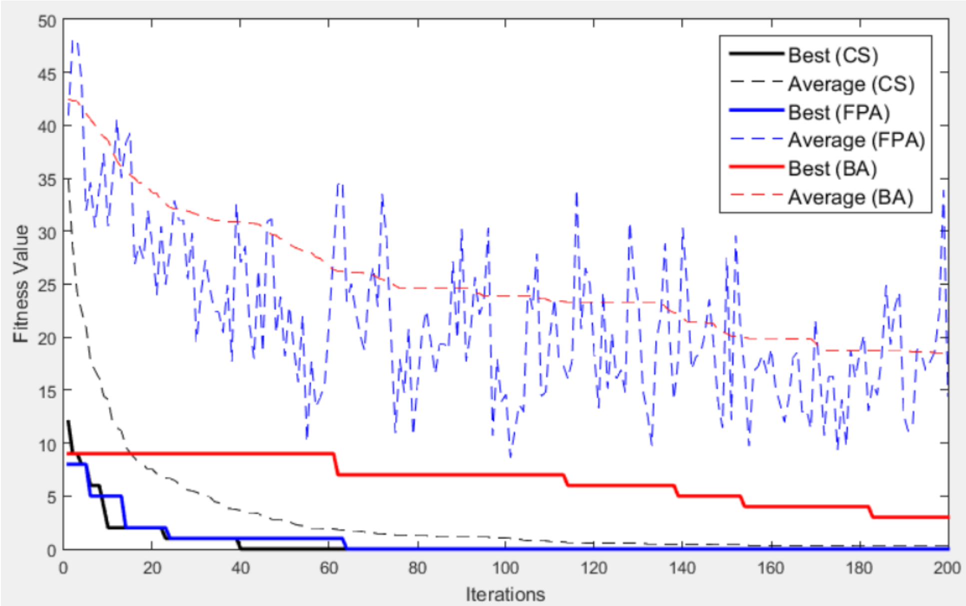
Figura 9: Fitness evaluations for (a) 150-user, 3-drone set; (b) 200-user, 4-drone set; (c) 300-user, 6-drone set



(a)



(b)



(c)

Source: Author

Since there are less UAV in less-populated search spaces, it is expected that the coverage area should be smaller, and thus, users connected to the array may take a hit.

Visual representations of drone positioning and SINR values for the 150-user set is seen as in Figure 10. Notice that users scattered throughout the user plane are represented in black dots, and their values may surpass the search space by a little - this is due to normal distribution not exactly having a limit on the values it may generate, but truncated normal distribution generation makes results become more proximate to the search space. The exact position of UAVs are represented by red dots. This coloring pattern is made standard in all SINR plots in this study.

For the SINR and positioning of UAVs in a 200-user, 4-drone set, results are shown in Figure 11.

And finally, the estimated values for SINR and UAV-array positioning in a 300-user, 6-drone set, results are seen in Figure 12.

In light of all obtained results, Table 4 denotes the error of the best fitness values for every algorithm and all user sets, the relative error C, and average time consumption in the simulation.

It is noticeable the amount of time consumption as more input variables are necessary to run the code. For instance, the time required to run 200 iterations of the cuckoo search for 150 users is little more than one and a half minute, but for 300 users it is around 6 minutes.

For reference, and according to values in Table 4, the best positioning solutions

Tabela 4: Best Fitness Outputs (Wi-Fi)

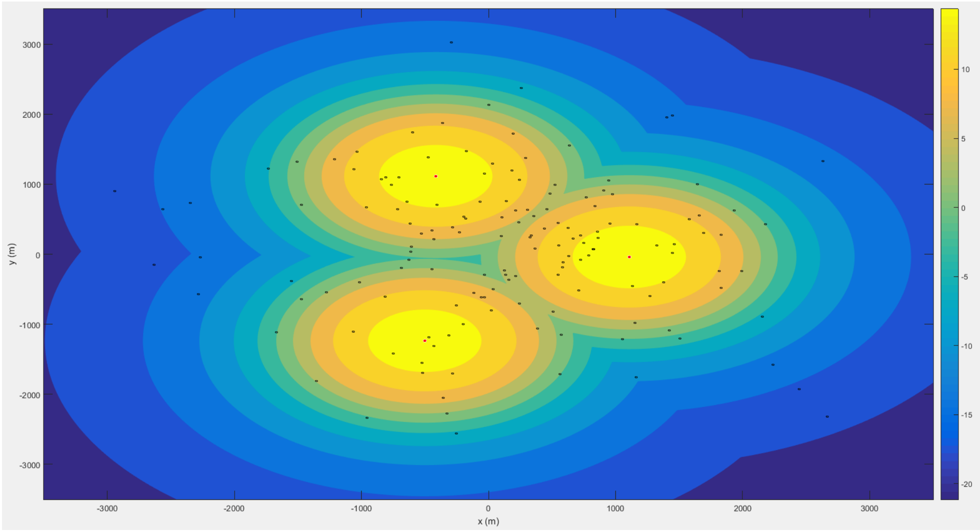
User Set	**Algorithm**	**Best Fitness**	**C**	**Run Time (s)**
$N_{users} = 150$	CS	21	14%	103.55
$N_{users} = 200$	CS	1	0.5%	159.899
$N_{users} = 300$	CS	0	0%	356.992
$N_{users} = 150$	FPA	21	14%	80.072
$N_{users} = 200$	FPA	2	1%	109.936
$N_{users} = 300$	FPA	0	0%	211.832
$N_{users} = 150$	BA	42	28%	55.56
$N_{users} = 200$	BA	12	6%	84.981
$N_{users} = 300$	BA	3	1.5%	188.436

Source: Author

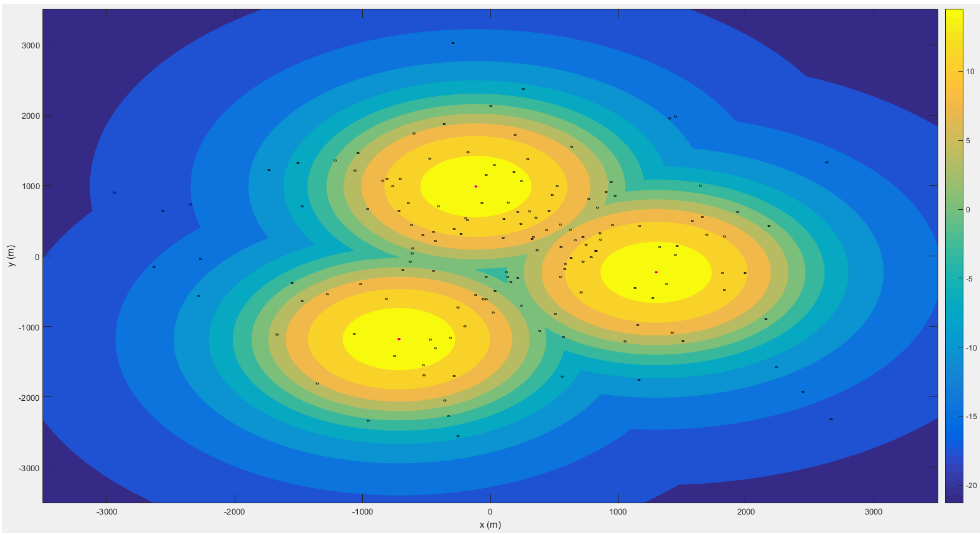
for each user set will be exposed below:

- For $N_{users} = 150$, the best optimizer is FPA. Best Solution (x, y, h) in meters: [-715.26, -1179.91, 392.68; 1297.54, -231.29, 399.99; -113.12, 986.49, 400]
- For $N_{users} = 200$, the best optimizer is CS. Best Solution (x, y, h) in meters: [3354.87, 1327.17, 397.58; 1408.97, -960.34, 399.98; -1740.58, 1934.03, 400; -1480.73, -1033.5, 400]
- For $N_{users} = 300$, the best optimizer is FPA. Best Solution (x, y, h) in meters: [-802.5, -1997.82, 400; 1018.64, 2000, 343.10; 1698.77, 926.86, 400; -822.28, 403.53, 399.99; 1835.55, -1879.63, 386.97; -1987.67, 1325.7, 395.45]

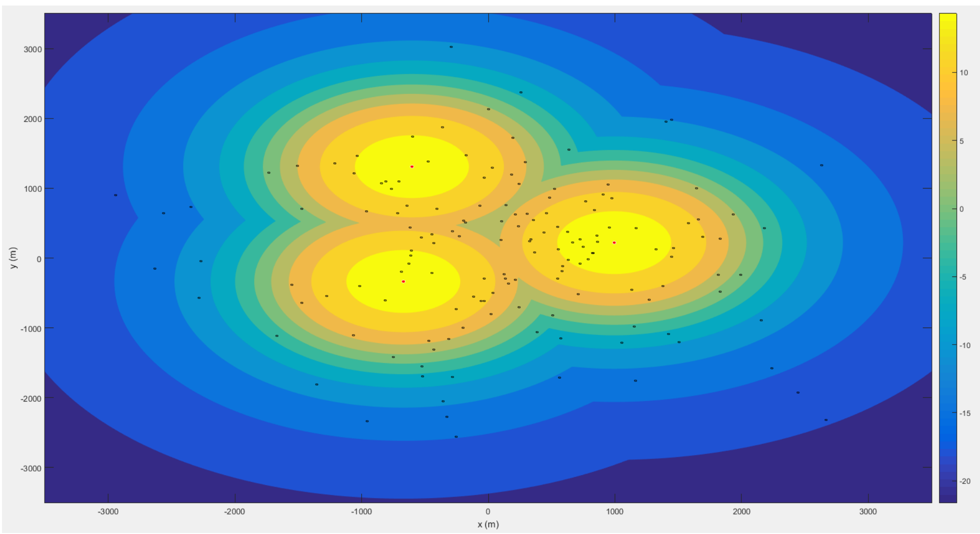
Figura 10: Drone positioning based on SINR values (see colorbar) for 150 users: (a) CS, (b) FPA and (c) BA



(a)



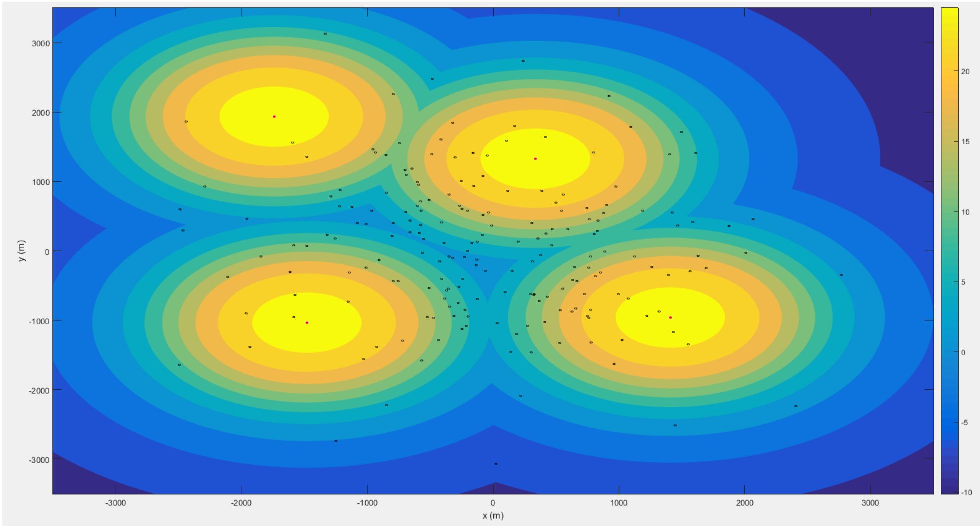
(b)



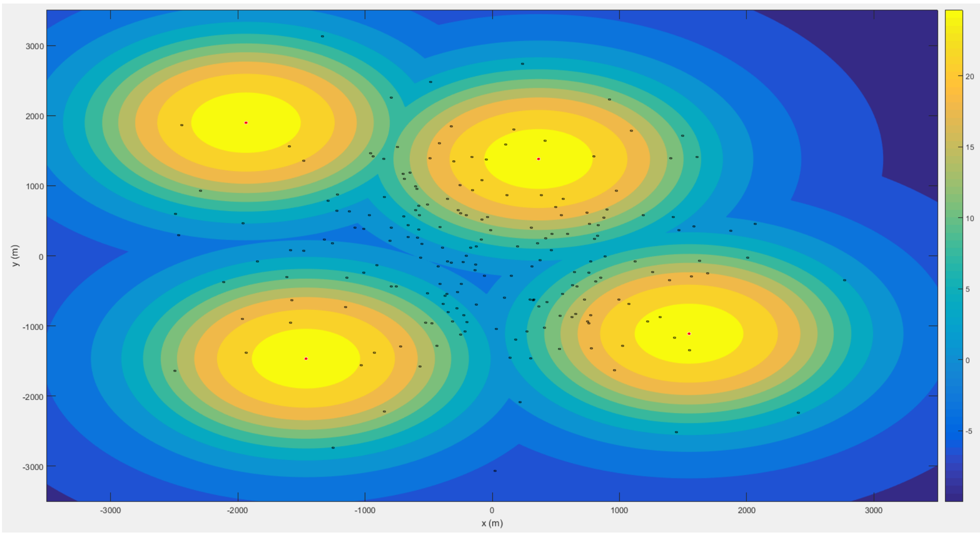
(c)

Source: Author

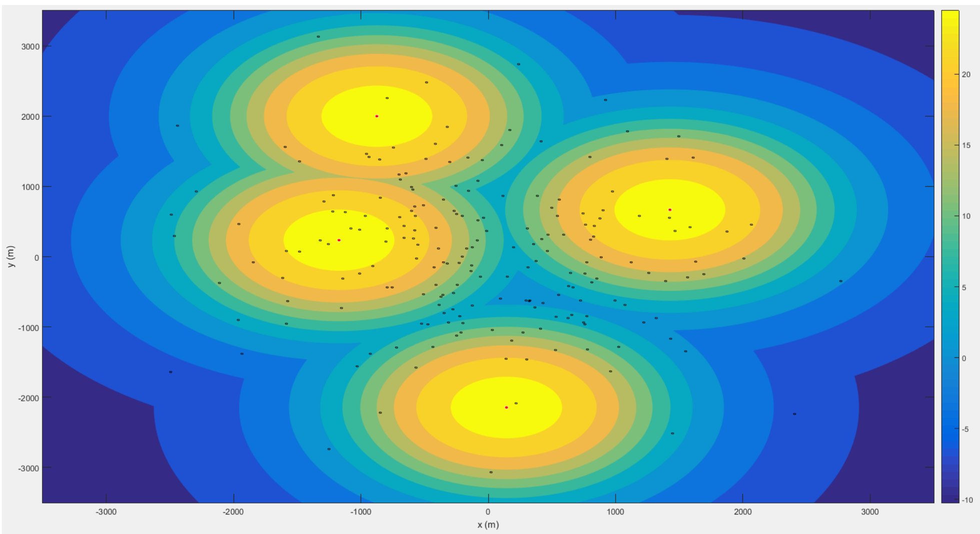
Figura 11: Drone positioning based on SINR values (see colorbar) for 200 users: (a) CS, (b) FPA and (c) BA



(a)



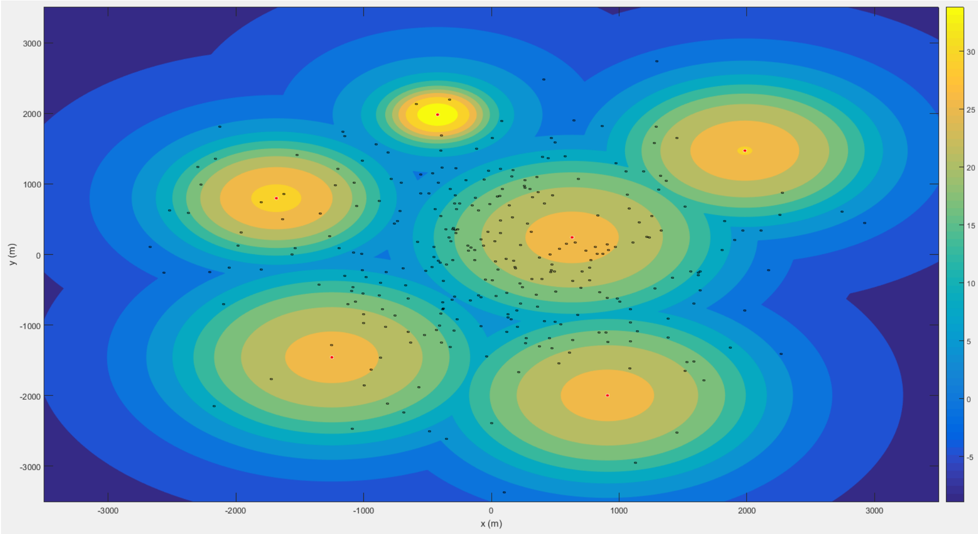
(b)



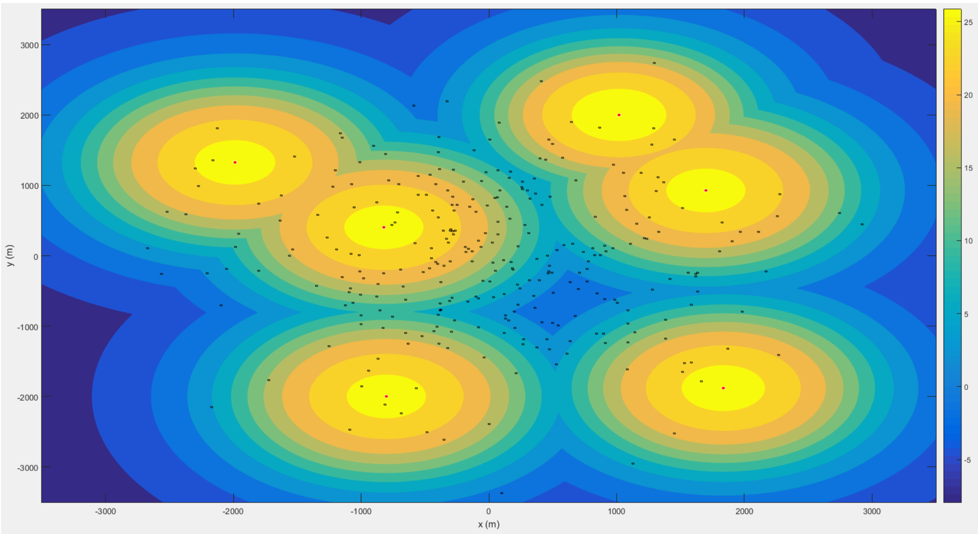
(c)

Source: Author

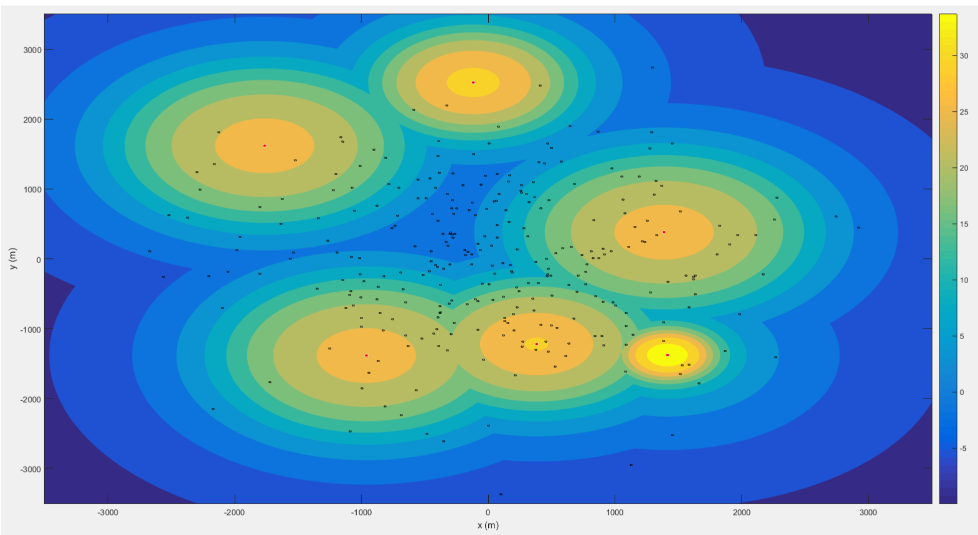
Figura 12: Drone positioning based on SINR values (see colorbar) for 300 users: (a) CS, (b) FPA and (c) BA



(a)



(b)



(c)

Source: Author

6.3 Theoretical LoRa Simulation

For the LoRa setting utilizing the equations specified in Section 5.2.1, and suggested by (MOZAFFARI et al., 2015), the three sets of network users N_{users} = [24; 40; 56] are to be utilized, resulting in N_{UAV} = [3; 5; 7], because the reference LoRa gateways chosen only support 8 consecutive users.

Users in LoRaWAN are the representation of sensors and/or other IoT devices that might be connected to its LPWAN network, be it mobile or fixed. This is why it is important to have an adaptive drone-array system to deal with such flexible traffic and movement of users.

Given that LoRa coverage area is considerably greater than what Wi-Fi can produce, the lower and upper bounds of the search space have been set as a plane of 256 km^2 (that is, 16x16 km) centered at (x, y) = (0, 0). This is a giant increase in coverage area compared to Wi-Fi, however, LoRa systems can keep connections up to a range between 2 km and 12 km depending on its SF. Given that the SF = 10 has been chosen for simulational purposes, it is safe to assume that the noise floor value should reflect the sensitivity of SF = 10 - which then would result in q = –132 dBm.

In Table 5, a list of relevant variables and constants that define the optimizations for the simulations of LoRaWAN are found.

Tabela 5: Variables and Constants of the Optimizations - Classic LoRa

Parameters	**Values**
Lower bounds (x, y and h) (m)	[-8000; -8000; 6]
Upper bounds (x, y and h) (m)	[8000; 8000; 60]
Number of iterations	500
Solutions per iterations	25
N_{users}	[24; 40; 56]
N_{UAV}	[3; 5; 7]
Switch Probability p (FPA)	0.3
Transmitted Frequency	915 MHz
Bandwidth	125 kHz
α	9.6
β	0.28
ζ_{LOS}	1 dB
ζ_{NLOS}	20 dB
q (SF = 10)	–132 dBm
G_t and G_r	0 dB
Transmitted Power (all UAVs)	20 dBm

Source: Author

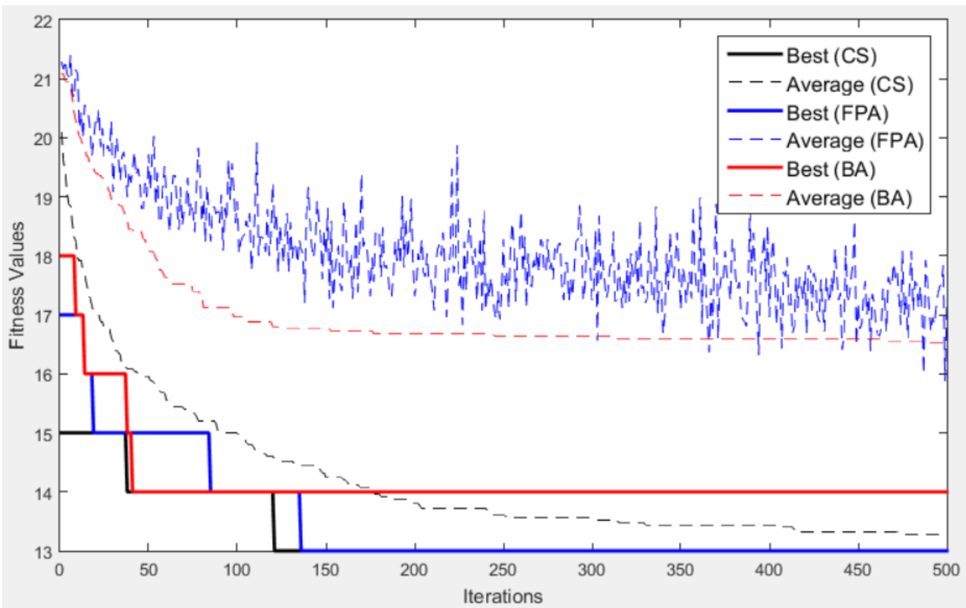
Drone heights have been set to between 6 to 60 meters, in order to correlate with data measured at the measurement campaigns (with set heights of [6; 24; 42; 60]).

The transmitted power has been kept to around 20 dBm, as many LoRa gateways can support this output to Tx antennas. For instance (SEMTECH, 2019b; SEMTECH, 2022) are a LoRa transceiver and a LoRa gateway, respectively, that can generate up to 22 dBm of Tx output. Propagation constants, therefore, have been kept equal to the ones in Wi-Fi simulation.

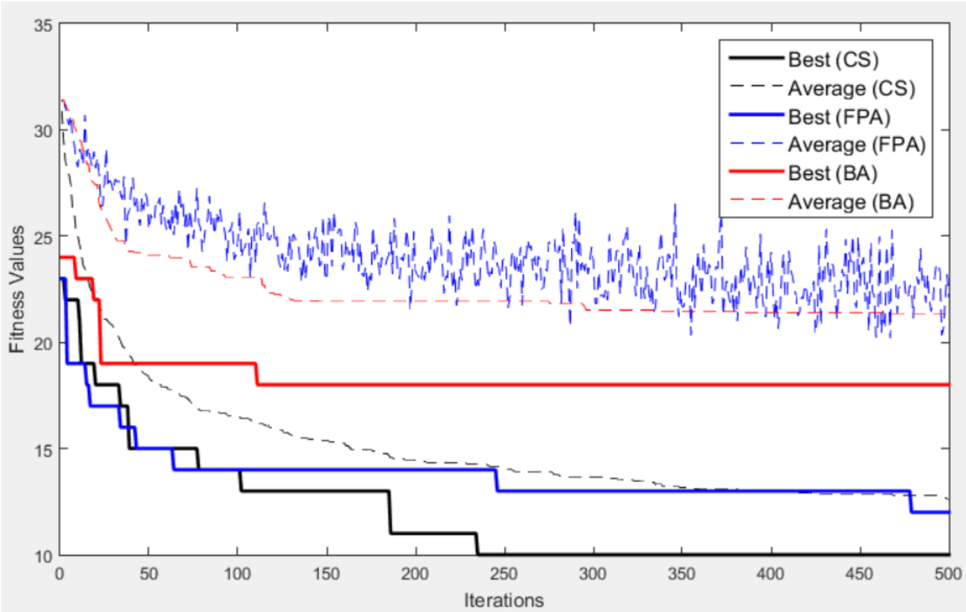
6.3.1 Theoretical LoRa Fitness and Propagation Results

The fitness evaluation of all algorithms, divided by user sets are present in Figure 9.

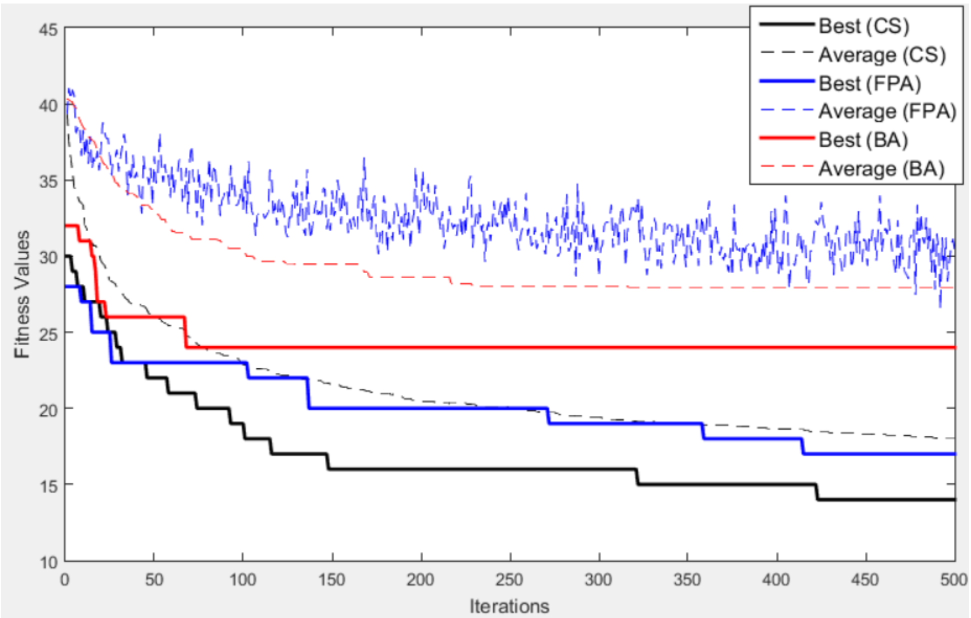
Figura 13: Fitness evaluations for (a) 24-user, 3-drone set; (b) 40-user, 5-drone set; (c) 56-user, 7-drone set



(a)



(b)



(c)

Source: Author

Visual representations of drone positioning and SINR values for the 24-user set is seen as in Figure 14.

For the SINR and positioning of UAVs in a 40-user, 5-drone set, results are shown in Figure 15.

And finally, the estimated values for SINR and UAV-array positioning in a 56-user, 7-drone set, results are seen in Figure 16.

Contrary as to what is observed in Wi-Fi simulations, in which peak SINR values are ever present in near distances but fade away quicker in far distances, LoRa propagation proves to possess a great range with lower high-SINR coverage in the near distances - but that preserves signal range throughout a much longer distance.

According to SF 10 range values given by LoRaWAN and Semtech themselves, each drone should be able to reach coverage areas of up to 8 km in diameter, which the UAVs in the pictures approximately represent.

Table 6 denotes the error of the best fitness values for every algorithm and all user sets, the relative error C, and average time cost of the simulation.

Tabela 6: Best Fitness Outputs (LoRa)

User Set	Algorithm	Best Fitness	C	Run Time (s)
$N_{users} = 24$	CS	13	54.17%	92.62
$N_{users} = 40$	CS	10	25%	114.146
$N_{users} = 56$	CS	14	25%	125.607
$N_{users} = 24$	FPA	13	54.17%	108.43
$N_{users} = 40$	FPA	14	35%	180.136
$N_{users} = 56$	FPA	16	28.57%	187.48
$N_{users} = 24$	BA	14	58.3%	60.189
$N_{users} = 40$	BA	18	45%	63.203
$N_{users} = 56$	BA	24	42.85%	71.417

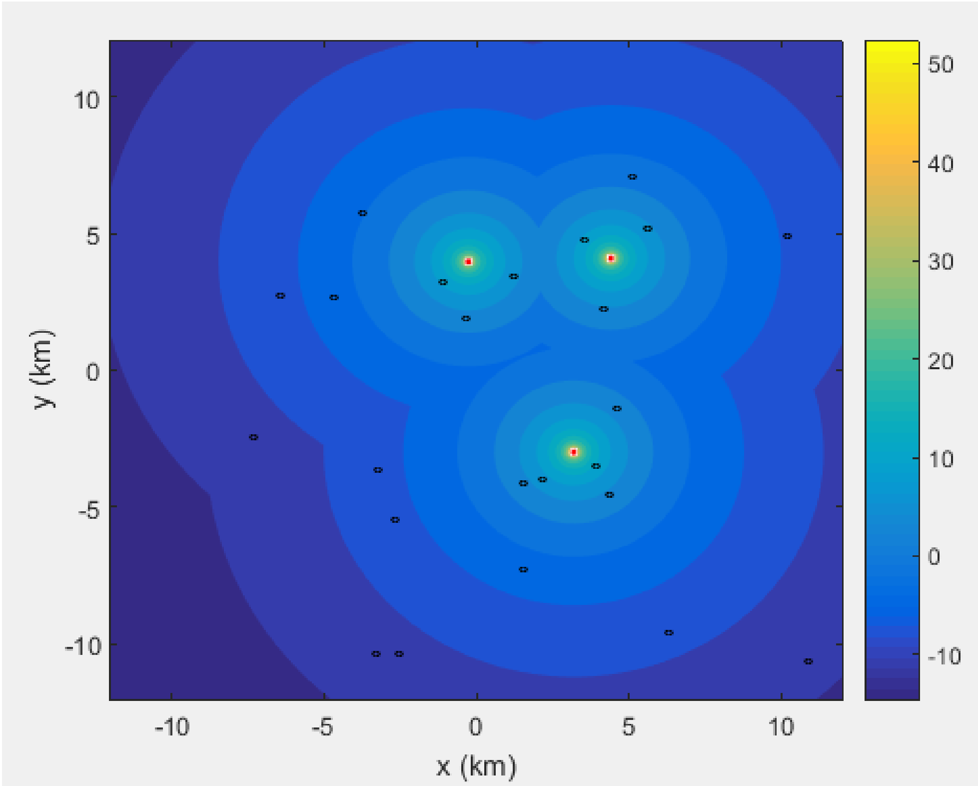
Source: Author

It is enough to see that these values are not satisfactory, and that in its limit of 200 iterations, this theoretical formula of LoRa falls short. Increasing the number of iterations may yield better fitness values, but this is of no use since the values found for the corrected LoRa model, in Section 6.4, converge faster and to better solutions.

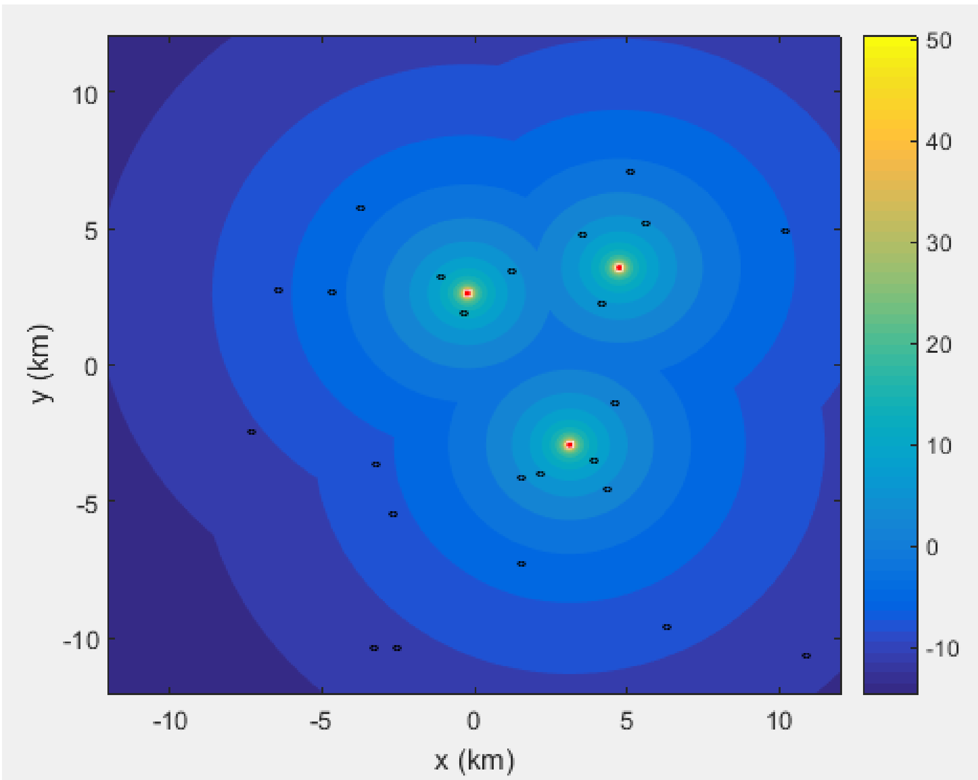
For reference, and according to values in Table 6, the best positioning solutions for each user set will be exposed below:

- For $N_{users} = 24$, the best optimizer is CS. Best Solution (x, y, h) in meters: [4409.54, 4089.43, 54.67; -265.7, 3969.24, 42.37; 3193.43, -2985.31, 55.87];
- For $N_{users} = 40$, the best optimizer is CS. Best Solution (x, y, h) in meters: [-5729.37, -2648.03, 7.05; 330.82, -1233.81, 6.35; 1400.12, 2342.55, 52.58; 637.72, -3963.64, 5.68; -3125.61, 2087.21, 6];
- For $N_{users} = 56$, the best optimizer is CS. Best Solution (x, y, h) in meters: [5477.3, -1463.32, 59; 1758.64, 1025.32, 58.21; -3726.87, -4968.69, 6; 1165.32, -1644.07, 9.04; 2221.12, -5488.63, 43.33; -3732.71, 1193.68, 6; -677.55, 7702.87, 6].

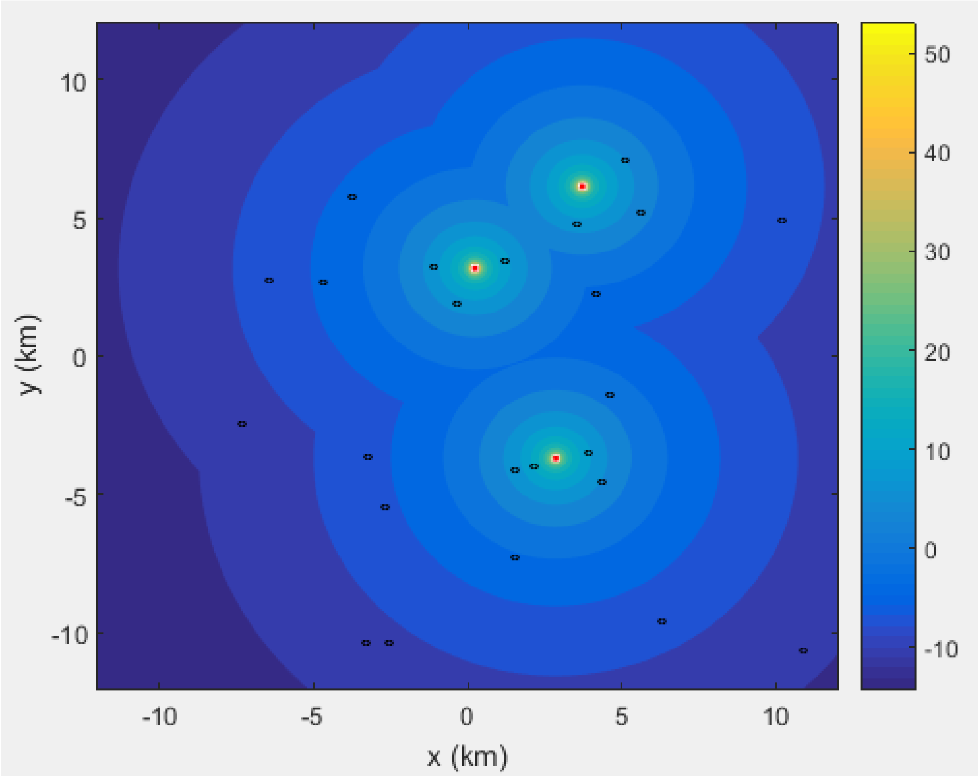
Figura 14: Drone positioning based on SINR values (see colorbar) for 24 users: (a) CS, (b) FPA and (c) BA



(a)

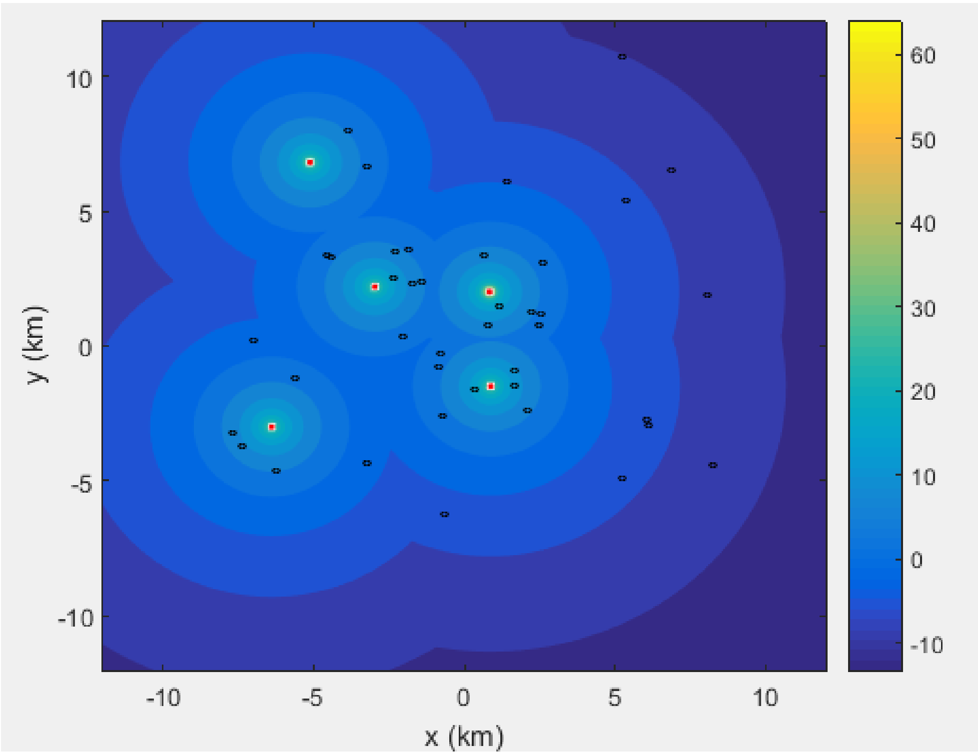


(b)



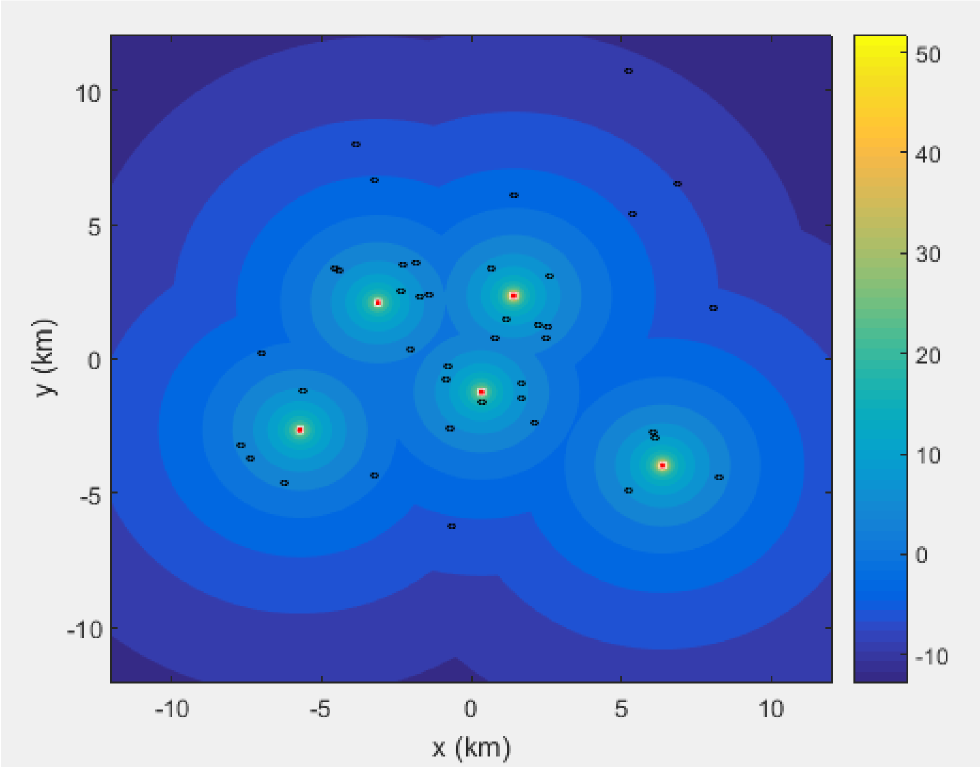
(c)

Source: Author

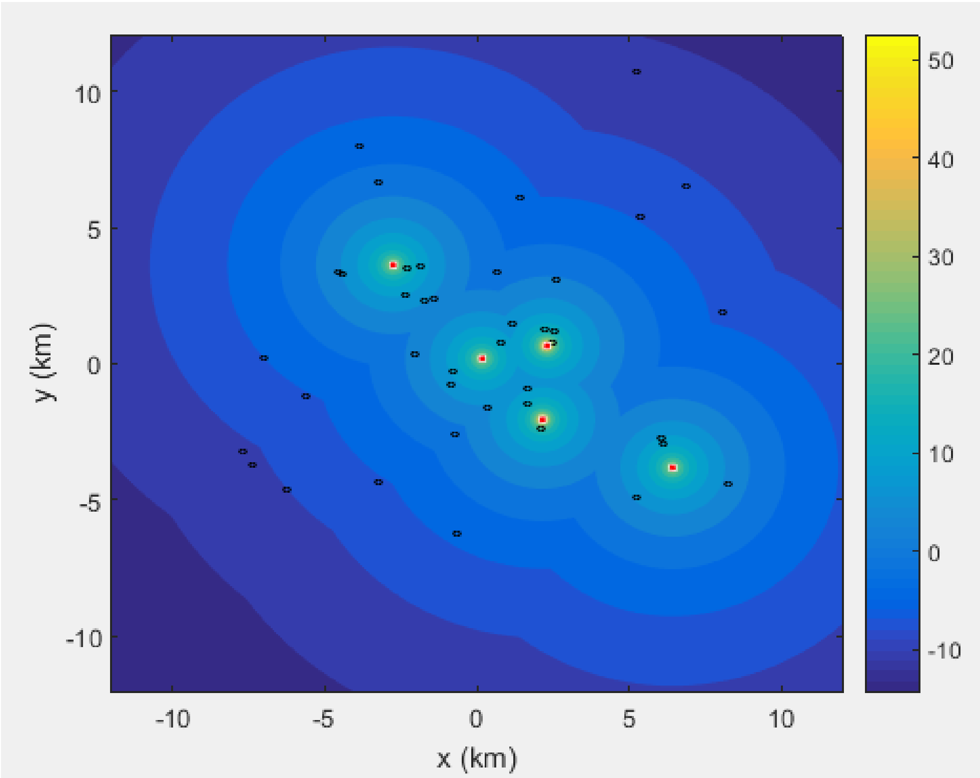


(b)

Figura 15: Drone positioning based on SINR values (see colorbar) for 40 users: (a) CS, (b) FPA and (c) BA



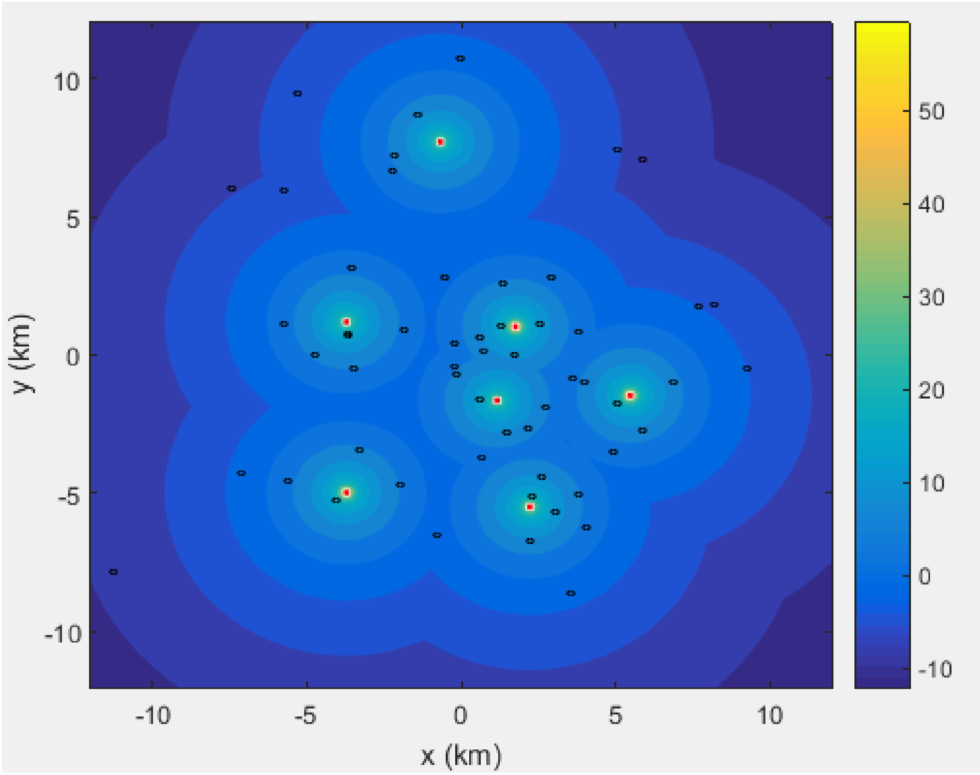
(a)



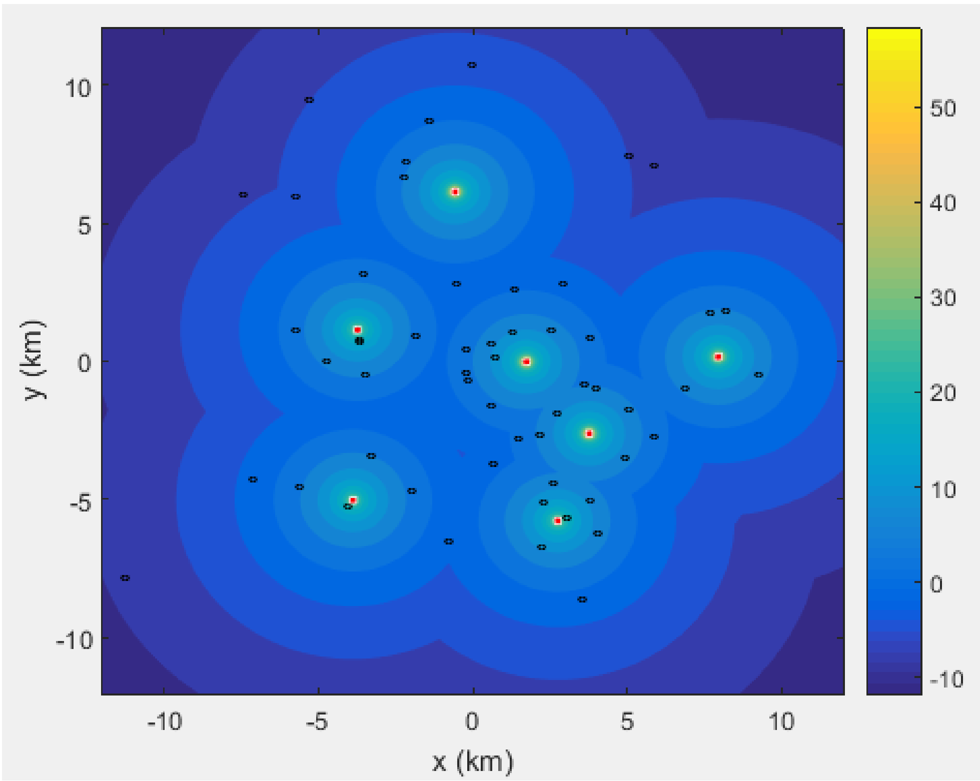
(c)

Source: Author

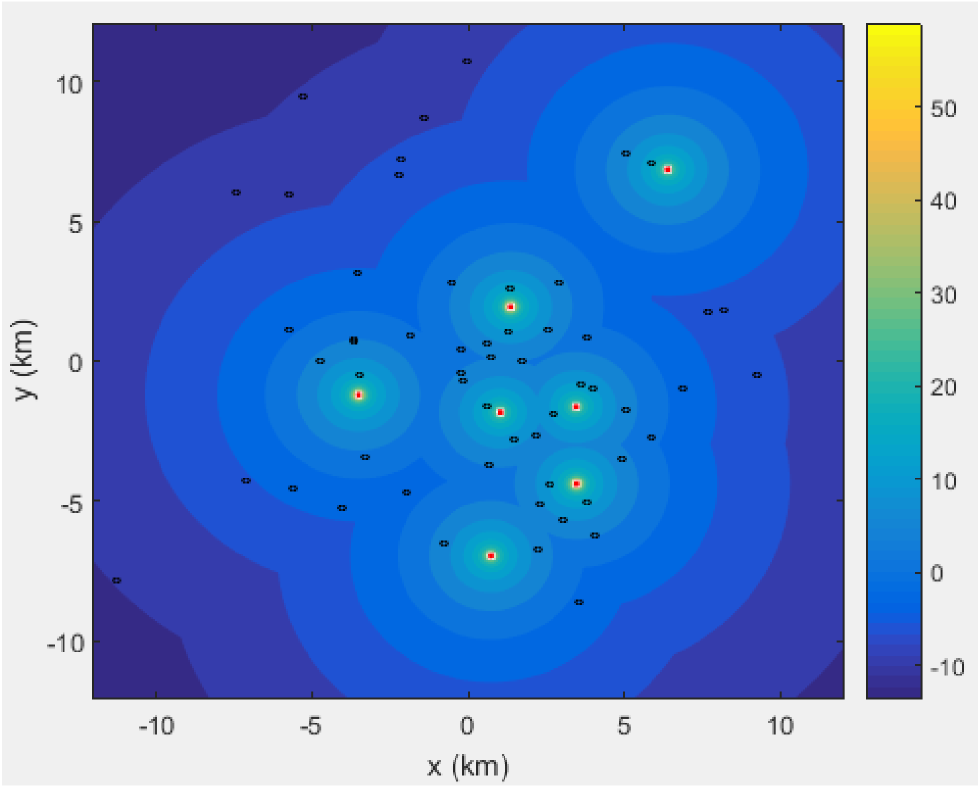
Figura 16: Drone positioning based on SINR values (see colorbar) for 56 users: (a) CS, (b) FPA and (c) BA



(a)



(b)



(c)

Source: Author

6.4 Empirically-Corrected LoRa Simulation

In this application of the simulation, many constant values for LoRa optimization seen in the last section are kept the same. The same values of height, search space bound area, and other details that build the boundaries of the optimization environment are equal to what is implemented in the theoretical LoRa optimization. The main difference, of course, is in the empirically-corrected propagation model that complements the LoRa optimization, putting into evidence the findings and mathematical modelings made in Section 5.2.3. The objective here is to keep as much proportionality between both types of LoRa optimization, in order to address if the LoRa propagation system with empirical adjustments can produce better results than the theoretical UAV equations.

In Table 7, a list of relevant variables and constants that define the optimizations for the simulations is compiled.

Tabela 7: Variables and Constants of the Optimizations - Empirical LoRa

Parameters	**Values**
Lower bounds (x, y and h) (m)	[-8000; -8000; 6]
Upper bounds (x, y and h) (m)	[8000; 8000; 60]
Number of iterations	500
Solutions per iterations	25
N_{users}	[24; 40; 56]
N_{UAV}	[3; 5; 7]
Switch Probability p (FPA)	0.3
Transmitted Frequency	915 MHz
Bandwidth	125 KHz
σ (SF10)	11.824
Path Loss Exponent (n)	4.6293
Reference Distance	94.06 m
Reference Path Loss	-76 dB
ζ_{LOS}	1 dB
ζ_{NLOS}	20 dB
q (SF = 10)	–132 dB
G_t and G_r	0 dB
Transmitted Power (all UAVs)	20 dBm

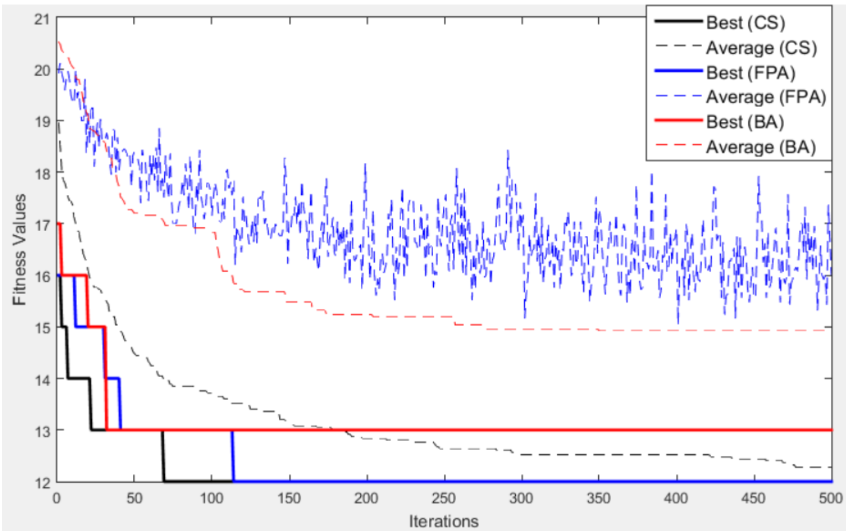
Source: Author

The standard deviation, path loss exponent and reference values are chosen for the LDPL modeling for the curve of SF 10 measurements (see Table 2), and thus these are the ones utilized for this implementation of simulations.

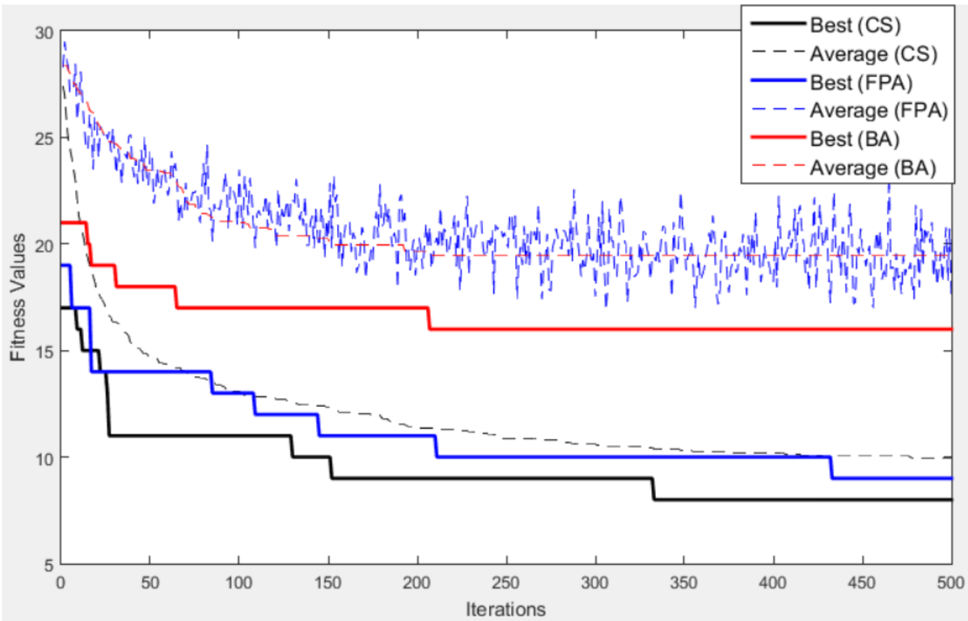
6.4.1 Empirically-Corrected LoRa Fitness and Propagation Results

The fitness evaluation of all algorithms, divided by user sets are present in Figure 17.

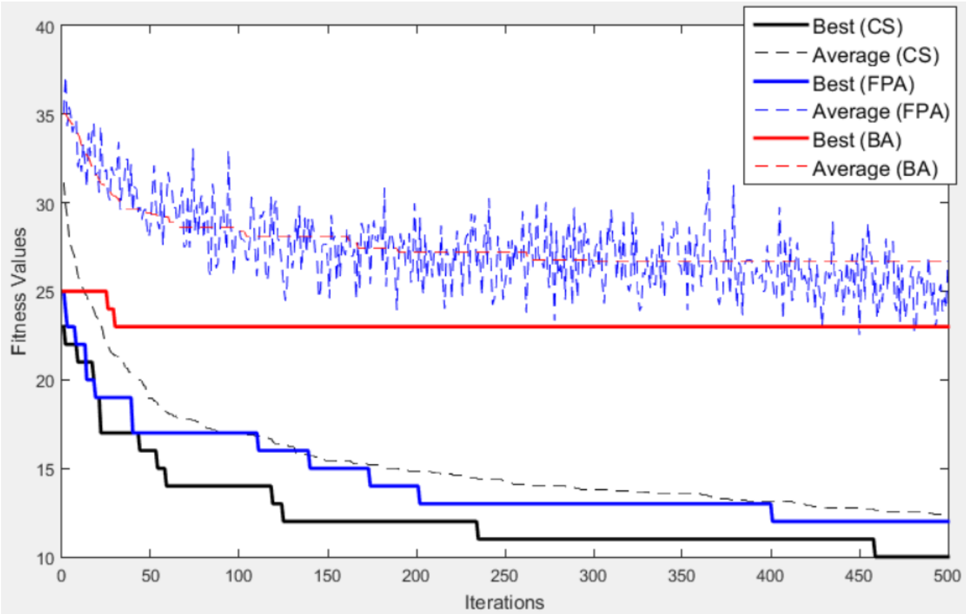
Figura 17: Fitness evaluations of Empirically-Corrected LoRa for (a) 24-user, 3-drone set; (b) 40-user, 5-drone set; (c) 56-user, 7-drone set



(a)



(b)



(c)

Source: Author

Visual representations of drone positioning and SINR values for the 24-user set is seen as in Figure 18.

For the SINR and positioning of UAVs in a 40-user, 5-drone set, results are shown in Figure 19.

Lastly, the estimated values for SINR and UAV-array positioning in a 56-user, 7-drone set, results are seen in Figure 20.

Table 8 denotes the error of the best fitness values for every algorithm and all user sets, the relative error C, and elapsed times of the simulations.

Tabela 8: Best Fitness Outputs (Empirically-Corrected LoRa)

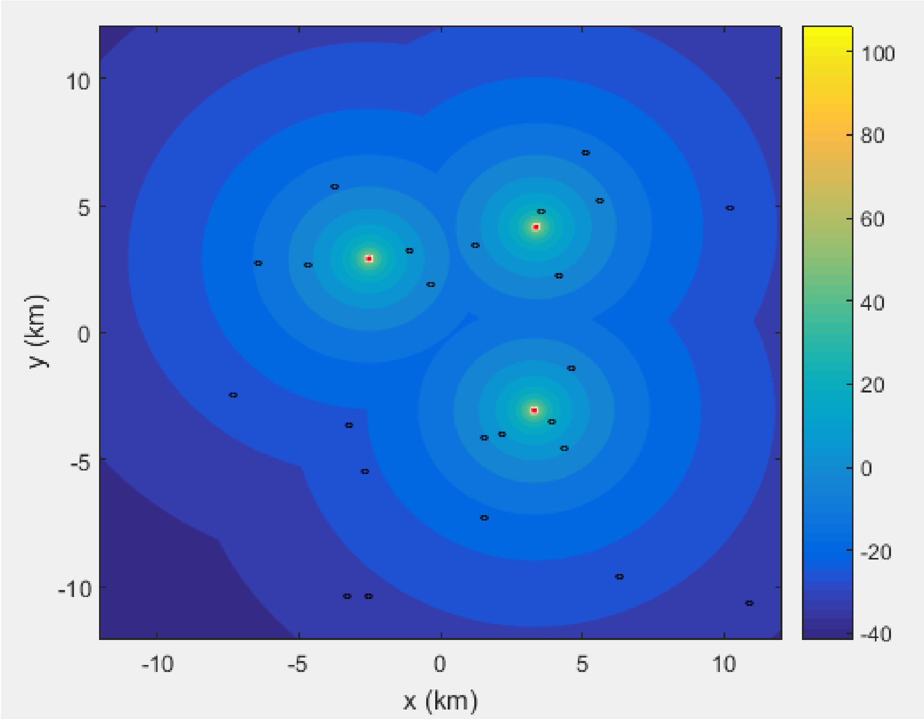
User Set	Algorithm	Best Fitness	C	Run Time (s)
$N_{users} = 24$	CS	12	50%	105.251
$N_{users} = 40$	CS	8	20%	111.262
$N_{users} = 56$	CS	10	17.85%	125.328
$N_{users} = 24$	FPA	12	50%	176.698
$N_{users} = 40$	FPA	9	22.5%	181.652
$N_{users} = 56$	FPA	11	19.64%	190.963
$N_{users} = 24$	BA	13	41.6%	54.815
$N_{users} = 40$	BA	16	40%	56.540
$N_{users} = 56$	BA	23	41.07%	64.041

Source: Author

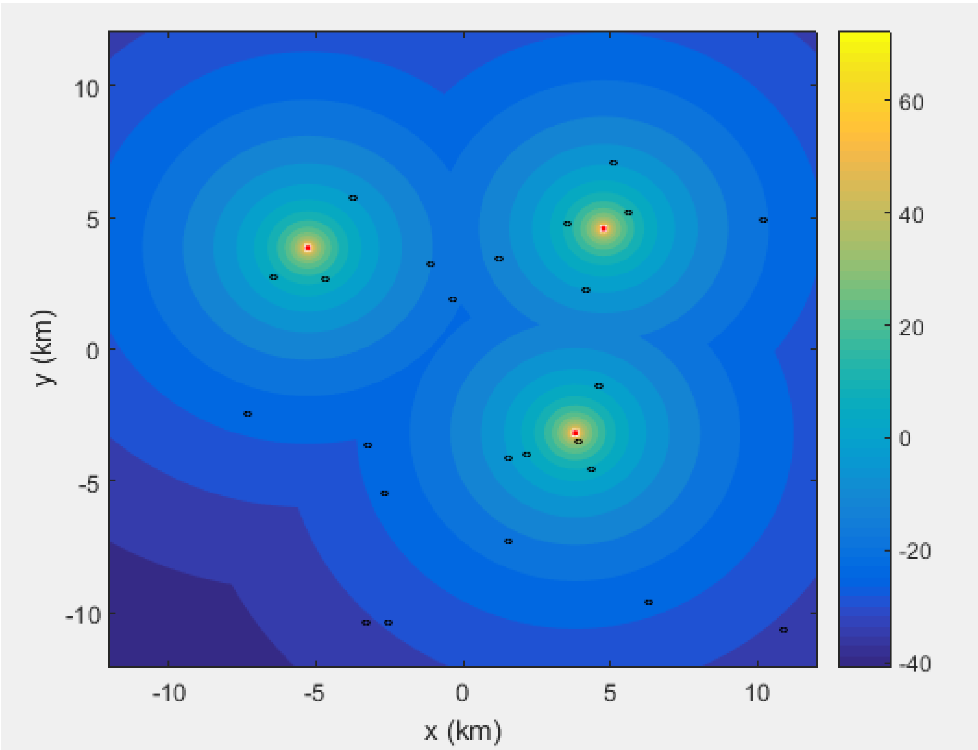
For reference, and according to values in Table 8, the best positioning solutions for each user set will be exposed below:

- For $N_{users} = 24$, the best optimizer is CS. Best Solution (x, y, h) in meters: [3299.09, -3055.51, 6; 3371.48, 4147.71, 18.15; -2533.08, 2906.48, 6];
- For $N_{users} = 40$, the best optimizer is CS. Best Solution (x, y, h) in meters: [-6678.2, -2181.07, 6; -2559.31, 2167.2, 6; 776.65, -558.94, 15.17; 5846.74, -4900.88, 59.95; 2964.94, 4613.36, 6];
- For $N_{users} = 56$, the best optimizer is CS. Best Solution (x, y, h) in meters: [-4377.95, 1046.65, 6; 1315.09, 1121.75, 6.8; -3988.44, -4820.87, 51.85; -4276.23, 7987.71, 59.36; 2781.01, -2518.95, 59.94; 3364, -7030.14, 6.2; 7701.87, -2431.09, 55.98].

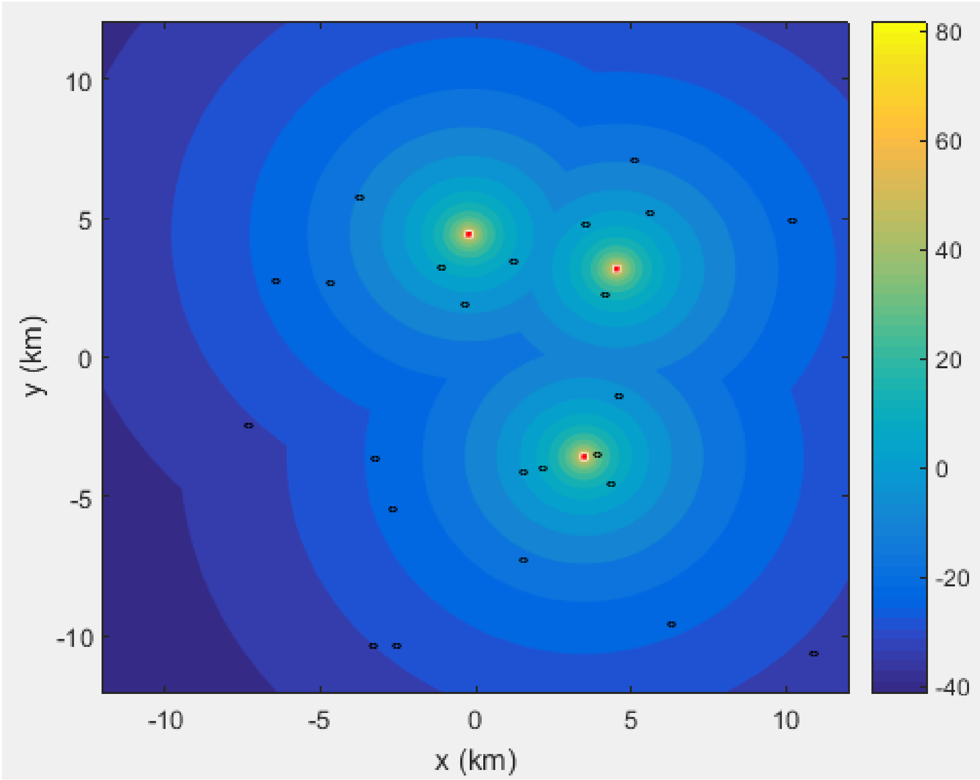
Figura 18: Drone positioning based on SINR values (see colorbar) for 24 users: (a) CS, (b) FPA and (c) BA



(a)



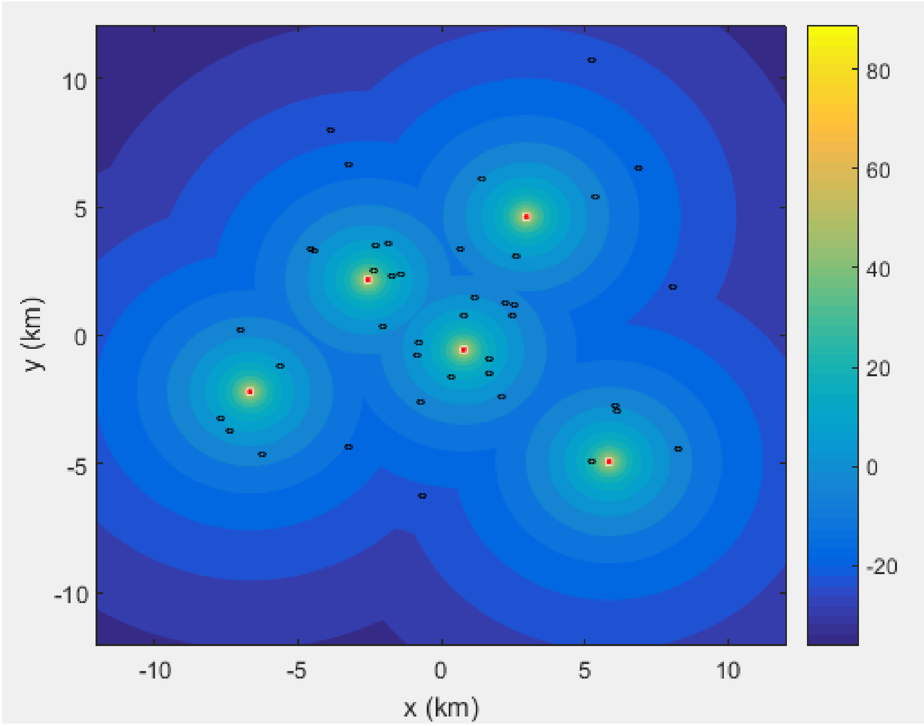
(b)



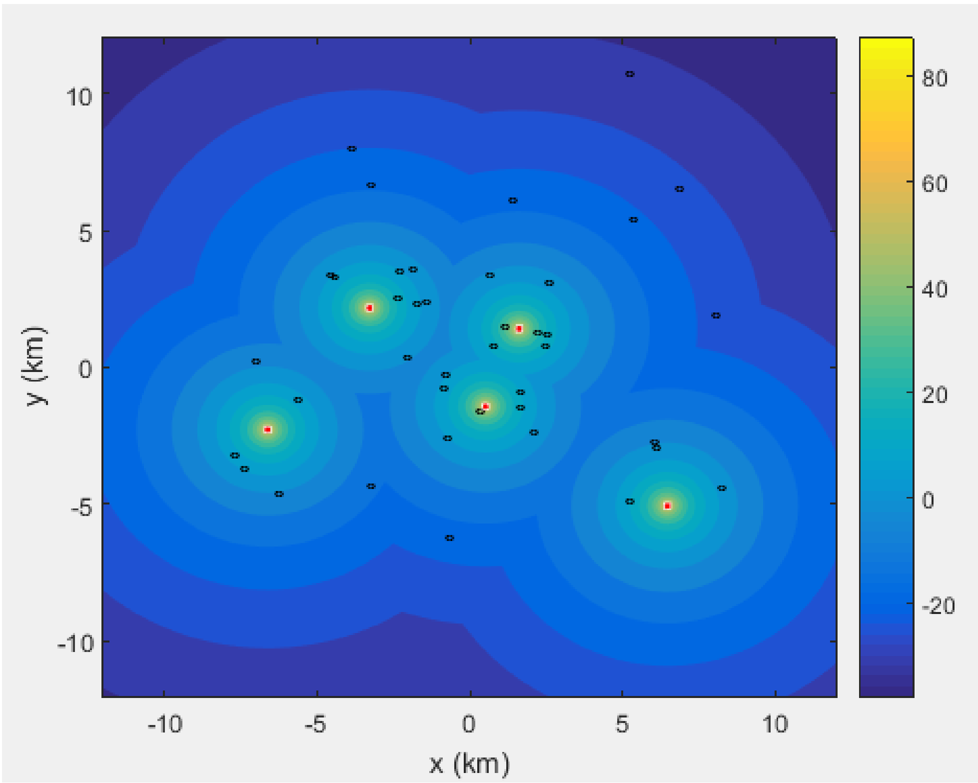
(c)

Source: Author

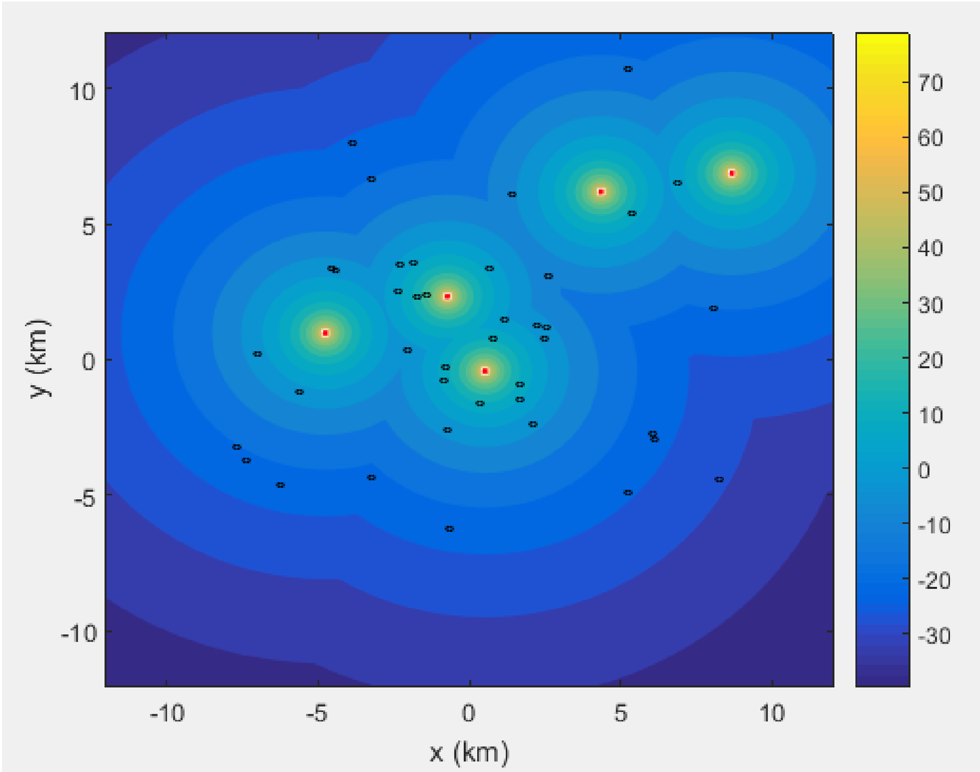
Figura 19: Drone positioning based on SINR values (see colorbar) for 40 users: (a) CS, (b) FPA and (c) BA



(a)



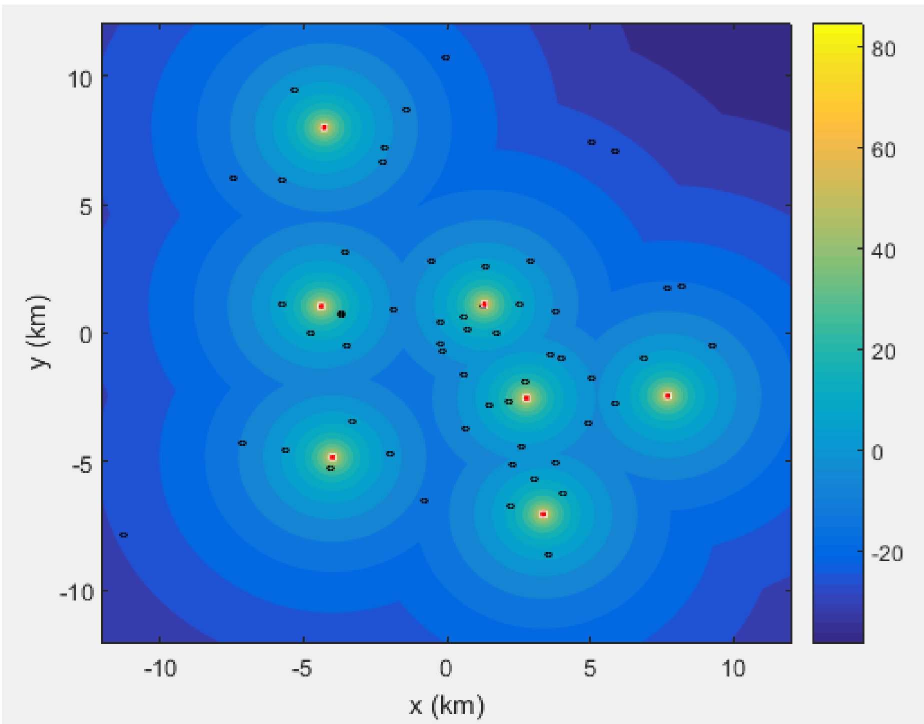
(b)



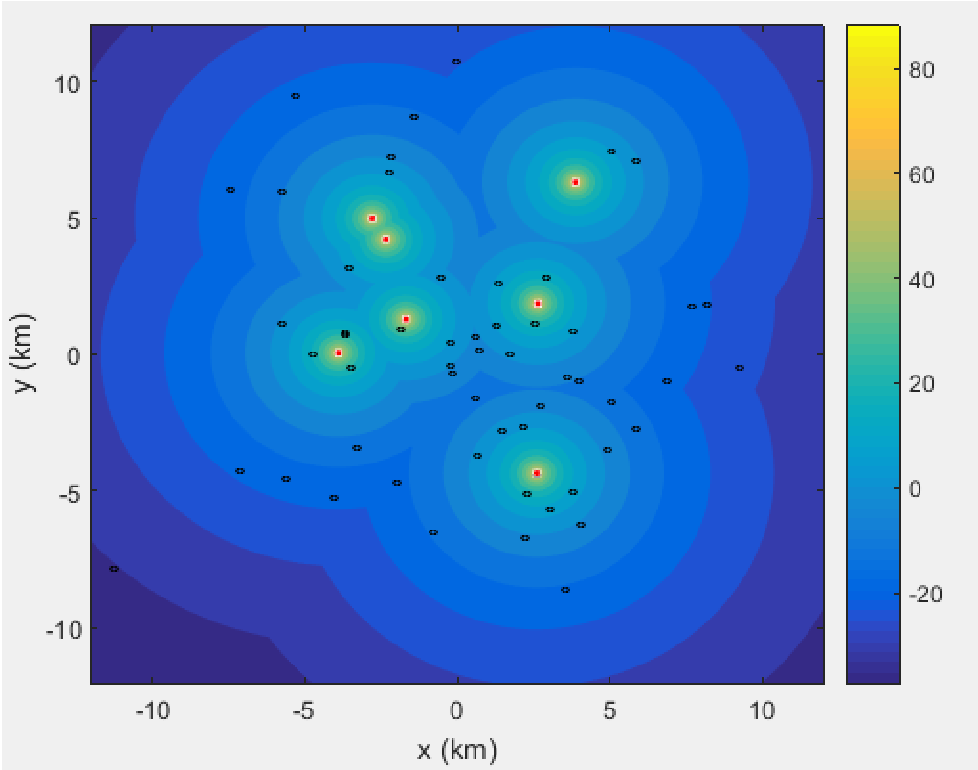
(c)

Source: Author

Figura 20: Drone positioning based on SINR values (see colorbar) for 56 users: (a) CS, (b) FPA and (c) BA

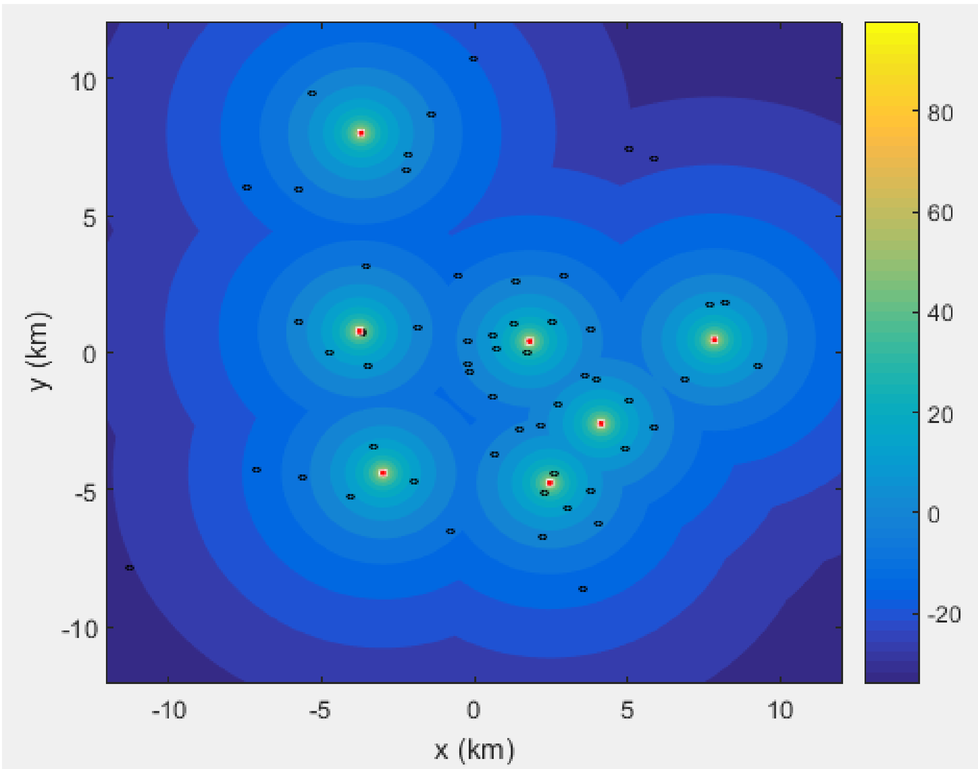


(a)



(c)

Source: Author



(b)

As it can be seen, contour lines do not differ greatly from the ones found in theoretical LoRa simulation. But the modeling does change, especially for the shorter distances in which the LDPL predicts higher SINR values. However, analyzing the images between both models, coverage area range shows little difference. And it is safe to conclude that the empirical model helps optimization algorithms to find more precise solutions.

6.5 Final Considerations

In this results section, several theoretical simulations have been conducted in two distinct search spaces (one for Wi-Fi, another for LoRa).

For all cases, it can be seen in the contour plots of SINR contained in this Chapter that the algorithms described in Section 5.3 have built a solid basis for the simulations. Drones tend to cluster into user-heavy areas and prefer those than going for hard-to-reach users, since this could mean less connected users overall. It can be observed, also, that they tend to not stay too close in order to avoid interference effects, being proximate only if power and interference values are low between UAVs.

Wi-Fi propagation simulation is kept to a very low level of disconnection of users, and results are satisfactory for 4 drones or more in the set search space. It is observed that, as drone quantity increases in a certain area, also greater is the coverage area, allowing for a user to connect to a more distant UAV base-station in case the one nearest to him is deemed at full capacity. Therefore, upscaling the number of drones, even with UAV-to-UAV interferences in mind and upscaling the number of users as well, is bound to produce better area of coverage results almost always.

It is important to notice that the greater amount of users present in the Wi-Fi simulations, compared to the LoRaWAN one, did impact significantly in running time of the algorithms - in some instances taking more than 5 minutes to reach an optimal value.

As for LoRa simulations, SF 10 values have been chosen as they have been figured to fit better into the simulational examples created in this study, as well as in its search space boundaries.

Comparing the results in Table 8 with Table 6, there is a glaring difference in the relative error of the results, as well as some time gain, all in favor of the empirically-corrected LoRa. It is safe to assume that, for all comparative simulations herein presented, that the empirical LoRa model produces better, more accurate results. Time cost for 500 iterations seems to tie across the board, with no propagation model being heavier than the other.

Also, around all simulations, the best optimized results all favor the cuckoo search algorithm, with flower pollination being a close second - sometimes matching the fitness provided by CS but losing on time cost. The bat algorithm did not produce the best results in this kind of problem, however, it is lightweight and for the rare cases in which it provides passable accuracy for much less computational time (see Wi-Fi results in 4, it might be proven useful.

7 Conclusion

In this dissertation, an extensive, simulation-heavy analysis and application of drone-array optimization has taken place. This has been made possible by extensive coding in MATLAB by the author, as well as much studying of not only optimizations via bioinspired algorithms, but also of propagation models for UAV wireless communication systems.

It is safe to presume that all objectives set in the introduction of this dissertation have been concluded. The proposal of a simulational environment to determine optimal positioning of drones in an array system in order to maximize a signal's area of coverage in MATLAB has been accomplished.

The objective of producing fast and accurate responses to the simulational problems proposed herein is achieved by the high computational speed of the bioinspired algorithm techniques. The algorithms are swift, rapid forms of optimization that are meant to be ported into an UAV micro-controller and used on-the-fly - or even by a remote control station with greater computational power, for faster results. With only a few seconds to obtain an optimal result for Wi-Fi networks and a few minutes for LoRaWANs it proves itself to be an adaptable method of optimally re-positioning drones to cover as many users as possible with acceptable (or higher) signal values.

Also, the objective of finding a way to relate measured data of LoRa for forested environments into a calculated model produces better results than purely theoretical, simulational data, as noticed in the improvements between LoRa simulations in this work.

Compared to the conference paper made by the author, published in (FERREIRA et al., 2021), where a raw version of said simulation environments was exposed, quality of results and methodology behind the study has improved massively. This paper only presented Wi-Fi, theoretical data, and a slightly inaccurate distance equation that eschewed results. Not only all of the flaws in the paper have been corrected, but its topics have been expanded upon. The addition of LoRa simulation, an estimate of calculated LoRa values via measured data, the crucial implementation of SINR and the enlargement of statistical data by simulating various user sets to prove the adaptability of the codes – all of this has been done in this dissertation.

7.0.1 Future Prospects

However, there are many prospects of improvement in this work. For instance, measured and calculated data in the empirically-corrected LoRa model still produced a somewhat large standard deviation, which tends to scatter calculated results further when on large distances. Finding a fix to this would, probably, improve the optimization process even more.

User association to drones in the array may actually be improved as well. Since the association method created and utilized in this dissertation only counts connection by maximum SINR, there is no way to guarantee with total certainty that a LoRa drone-array guided by the algorithms present herein shall connect up to exactly 8 users per drone. So, the optimization techniques depend upon a third-party code or a native LoRa support that can aid in these situations of "user overflow", or inability to connect due to reaching the maximum capacity of a drone.

And to sum up to ideas of improvement, a way to better automatize transmitted power values would be of interest to the literature. For instance, if the code is adapted to minimize the necessary transmitted power, that would mean more battery saving for UAVs, transmitters, transceivers and gateways. Energy efficiency to improve the air-time of UAVs is an extensively studied topic throughout the literature, and this could be a welcome addition.

One general manner in which the transmitted power can be optimized is by attaching it with some equation or mathematical condition as on output in the objective function - but as of this moment the implementation necessary to accomplish this is unknown to the author.

As prospects for the future, the author wishes to continue studying on the subject of bioinspired optimization and finding ways to attach this optimization knowledge to applications in dynamic networks such as the ones found in UAV arrays. The contents of this dissertation shall be compiled into a journal paper, by will of the author and the recommendation of his academic advisor.

7.0.2 Publications

Some of the publications which the author of this dissertation has penned or participated as coauthor are listed below:

1. FERREIRA, F.H. et al. Tri-Band, Stable and Compact Patch Frequency Selective Surface Optimized via Hybrid Bioinspired Computing for Applications at 2.4, 3.5 and 5.8 GHz. Journal of Microwaves, Optoelectronics and Electromagnetic Applications. To Be Published. Accepted: June 25th, 2021.

2. FERREIRA, F.H. et al. Positioning Optimization of Base-Station Drone Arrays via Bioinspired Computing Techniques. In: XI Conferência Nacional em Comunicações, Redes e Segurança da Informação (ENCOM 2021), 2021, online.

3. ALCANTARA NETO, M. C. ; ARAUJO, J. P. L. ; BARROS, F. J. B. ; FERREIRA, F. HENRY C. S. ; CASTRO, B. S. L. ; MOTA, R. J. S. ; CAVALCANTE, G. P. S. . Design and Synthesis of an Ultra Wide Band FSS for mm-Wave Application via General Regression Neural Network and Multiobjective Bat Algorithm. Journal of Microwaves, Optoelectronics and Electromagnetic Applications, v. 18, p. 530-544, 2019.

4. LIMA, W. ; NOGUEIRA, C. ; FERREIRA, F. HENRY C. S. ; BARROS, F. J. B. ; ALCANTARA NETO, M. C. ; ARAUJO, J. P. L. ; CAVALCANTE, G. P. S. . Hybrid and Bioinspired Computational Optimization Techniques for the Design and Synthesis of Multilayer FSS. In: International Microwave and Optoelectronics Conference (IMOC 2019), 2019, Aveiro, Portugal.

5. LOPES, A. ; ROCHA, L. ; FERREIRA, F. HENRY C. S. ; BATALHA, I. ; GOMES, C. ; BARROS, F. J. B. ; CARVALHO, A. . Large-Scale Propagation Modeling in Corridors at the 10 GHz Frequency Spectrum. In: International Microwave and Optoelectronics Conference (IMOC 2019), 2019, Aveiro, Portugal.

References

AGGARWAL, K.; GOYAL, A. Particle swarm optimization based uav for disaster management. In: IEEE. *2021 IEEE 5th Advanced Information Technology, Electronic and Automation Control Conference (IAEAC)*. [S.l.], 2021. v. 5, p. 1235–1238. Citado na página 6.

AICARDI, I. et al. Monitoring post-fire forest recovery using multitemporal digital surface models generated from different platforms. *EARSeL eProceedings*, 2016. v. 15, n. 1, p. 1–8, 2016. Citado na página 7.

AL-TURJMAN, F. et al. Enhanced deployment strategy for the 5g drone-bs using artificial intelligence. *IEEE Access*, 2019. IEEE, v. 7, p. 75999–76008, 2019. Citado na página 5.

ALDAHDOUH, K. A. et al. A survey of 5g emerging wireless technologies featuring lorawan, sigfox, nb-iot and lte-m. In: IEEE. *2019 International Conference on Wireless Communications Signal Processing and Networking (WiSPNET)*. [S.l.], 2019. p. 561–566. Citado na página 10.

BORGES, L. M.; VELEZ, F. J.; LEBRES, A. S. Survey on the characterization and classification of wireless sensor network applications. *IEEE Communications Surveys & Tutorials*, 2014. IEEE, v. 16, n. 4, p. 1860–1890, 2014. Citado na página 1.

BORKAR, S. R. Long-term evolution for machines (lte-m). In: *LPWAN Technologies for IoT and M2M Applications*. [S.l.]: Elsevier, 2020. p. 145–166. Citado na página 10.

CARDOSO, E. H. S. *Novo Autômato Celular Multicamada para Posicionamento de Celulas Voadoras em Redes Moveis 5G Auto-organizaveis*. Tese (Doutorado) — Doctorate Thesis, UFPA, ITEC, PPGEE, 2015. Citado na página 25.

CARUSO, A. et al. Collection of data with drones in precision agriculture: Analytical model and lora case study. *IEEE Internet of Things Journal*, 2021. IEEE, 2021. Citado na página 4.

CHALL, R. E.; LAHOUD, S.; HELOU, M. E. Lorawan network: Radio propagation models and performance evaluation in various environments in lebanon. *IEEE Internet of Things Journal*, 2019. IEEE, v. 6, n. 2, p. 2366–2378, 2019. Citado na página 27.

CID, E. L.; ALEJOS, A. V.; SANCHEZ, M. G. Signaling through scattered vegetation: Empirical loss modeling for low elevation angle satellite paths obstructed by isolated thin trees. *ieee vehicular technology magazine*, 2016. IEEE, v. 11, n. 3, p. 22–28, 2016. Citado na página 7.

CRYS, T.; HWEE-PINK, T. Evaluation of sigfox lpwan for sensor-enabled homes to identify at risk community dwelling seniors. In: IEEE. *2019 IEEE 44th Conference on Local Computer Networks (LCN)*. [S.l.], 2019. p. 26–33. Citado na página 9.

DAMBAL, V. A. et al. Improving lora signal coverage in urban and sub-urban environments with uavs. In: IEEE. *2019 International Workshop on Antenna Technology (iWAT)*. [S.l.], 2019. p. 210–213. Citado na página 4.

DAVE, M. et al. Versatile multipurpose crashproof uav: Machine learning and iot approach. In: IEEE. *2020 International Conference for Emerging Technology (INCET)*. [S.l.], 2020. p. 1–7. Citado na página 5.

DELAFONTAINE, V. et al. Drone-aided localization in lora iot networks. *arXiv preprint arXiv:2004.03852*, 2020. 2020. Citado na página 4.

DUANGSUWAN, S.; MAW, M. M. Comparison of path loss prediction models for uav and iot air-to-ground communication system in rural precision farming environment. *J. Commun.*, 2021. v. 16, n. 2, p. 60–66, 2021. Citado na página 7.

FERREIRA, F. H. C. S. et al. Positioning optimization of base-station drone arrays via bioinspired computing techniques. In: IECOM. *XI Conferência Nacional em Comunicações, Redes e Segurança da Informação - ENCOM 2021*. [S.l.], 2021. Citado 3 vezes nas páginas 23, 38 e 66.

GHAZALI, M. H. M.; TEOH, K.; RAHIMAN, W. A systematic review of real-time deployments of uav-based lora communication network. *IEEE Access*, 2021. IEEE, 2021. Citado na página 5.

GOLDSMITH, A. *Wireless communications*. [S.l.]: Cambridge university press, 2005. Citado 2 vezes nas páginas 25 e 28.

HENG, Z.; HONGWEI, W. Application test of lte-m in pis for urban rail transit system. *China Safety Science Journal*, 2019. v. 29, n. S2, p. 47, 2019. Citado na página 10.

KALANTARI, E.; YANIKOMEROGLU, H.; YONGACOGLU, A. On the number and 3d placement of drone base stations in wireless cellular networks. In: IEEE. *2016 IEEE 84th Vehicular Technology Conference (VTC-Fall)*. [S.l.], 2016. p. 1–6. Citado na página 5.

KHANDURI, R.; RATTAN, S. Performance comparison analysis between ieee 802.11 a/b/g/n standards. *International Journal of Computer Applications*, 2013. Citeseer, v. 78, n. 1, p. 13–20, 2013. Citado na página 25.

KHAWAJA, W.; GUVENC, I.; MATOLAK, D. Uwb channel sounding and modeling for uav air-to-ground propagation channels. In: IEEE. *2016 IEEE global communications conference (GLOBECOM)*. [S.l.], 2016. p. 1–7. Citado na página 7.

KHAWAJA, W. et al. A survey of air-to-ground propagation channel modeling for unmanned aerial vehicles. *IEEE Communications Surveys & Tutorials*, 2019. IEEE, v. 21, n. 3, p. 2361–2391, 2019. Citado na página 7.

LAVRIC, A.; POPA, V. Internet of things and lora™ low-power wide-area networks: a survey. In: IEEE. *2017 International Symposium on Signals, Circuits and Systems (ISSCS)*. [S.l.], 2017. p. 1–5. Citado 3 vezes nas páginas 1, 11 e 12.

MARTINEZ, B. et al. Exploring the performance boundaries of nb-iot. *IEEE Internet of Things Journal*, 2019. IEEE, v. 6, n. 3, p. 5702–5712, 2019. Citado na página 10.

MICHEZ, A. et al. Classification of riparian forest species and health condition using multi-temporal and hyperspatial imagery from unmanned aerial system. *Environmental monitoring and assessment*, 2016. Springer, v. 188, n. 3, p. 146, 2016. Citado na página 7.

MOZAFFARI, M. et al. Drone small cells in the clouds: Design, deployment and performance analysis. In: IEEE. *2015 IEEE global communications conference (GLOBECOM)*. [S.l.], 2015. p. 1–6. Citado 5 vezes nas páginas 23, 24, 29, 35 e 46.

MOZAFFARI, M. et al. A tutorial on uavs for wireless networks: Applications, challenges, and open problems. *IEEE communications surveys & tutorials*, 2019. IEEE, v. 21, n. 3, p. 2334–2360, 2019. Citado na página 6.

NETO, M. C. D. A. *SÍNTESE DE SUPERFÍCIES SELETIVAS DE FREQUÊNCIA PARA MICRO-ONDAS UTILIZANDO OTIMIZAÇÃO MULTIOBJETIVO BIOINSPIRADA*. Tese (Doutorado) — Doctorate Thesis, UFPA, ITEC, PPGEE, 2015. Citado na página 15.

NEWHALL, W. G. et al. Wideband air-to-ground radio channel measurements using an antenna array at 2 ghz for low-altitude operations. In: IEEE. *IEEE Military Communications Conference, 2003. MILCOM 2003.* [S.l.], 2003. v. 2, p. 1422–1427. Citado na página 7.

PIERZCHAŁA, M.; TALBOT, B.; ASTRUP, R. Estimating soil displacement from timber extraction trails in steep terrain: application of an unmanned aircraft for 3d modelling. *Forests*, 2014. Multidisciplinary Digital Publishing Institute, v. 5, n. 6, p. 1212–1223, 2014. Citado na página 7.

RAHMADHANI, A. et al. Lorawan as secondary telemetry communication system for drone delivery. In: IEEE. *2018 IEEE International Conference on Internet of Things and Intelligence System (IOTAIS)*. [S.l.], 2018. p. 116–122. Citado na página 5.

RAJASHREE, S.; RAJESH, A.; ANNEPU, V. Localization of co-operative sensor network and uav using bio-inspired optimization technique. In: IEEE. *2019 International Conference on Vision Towards Emerging Trends in Communication and Networking (ViTECoN)*. [S.l.], 2019. p. 1–6. Citado na página 6.

RIBEIRO, G. G. et al. An outdoor localization system based on sigfox. In: IEEE. *2018 IEEE 87th Vehicular Technology Conference (VTC Spring)*. [S.l.], 2018. p. 1–5. Citado na página 9.

SEMTECH. *LoRa Modulation Basics*. [S.l.], 2015. Disponível em: <https://web.archive-.org/web/20190718200516/https://www.semtech.com/uploads/documents/an1200.22-.pdf>. Citado 3 vezes nas páginas 11, 25 e 26.

SEMTECH. *LoRa and LoRaWAN: A Technical Overview*. [S.l.], 2019. Disponível em: <https://lora-developers.semtech.com/documentation/tech-papers-and-guides/lora-and-lorawan/>. Citado 3 vezes nas páginas 11, 12 e 26.

SEMTECH. *SX1272/73 - 860 MHz to 1020 MHz Low Power Long Range Transceiver*. [S.l.], 2019. Disponível em: <https://semtech.my.salesforce.com/sfc/p/E0000000JelG/a-/440000001NCE/v_VBhk1IolDgxwwnOpcS_vTFxPfSEPQbuneK3mWsXlU>. Citado na página 47.

SEMTECH. *LoRa Core™ PicoCell Gateway, SX1308, 915MHz.* [S.l.], 2022. Disponível em: <https://www.semtech.com/products/wireless-rf/lora-core/sx1308p915gw>. Citado na página 47.

SER, J. D. et al. Bio-inspired computation: Where we stand and what's next. *Swarm and Evolutionary Computation*, 2019. Elsevier, v. 48, p. 220–250, 2019. Citado na página 21.

SHEHAB, M.; KHADER, A. T.; AL-BETAR, M. A. A survey on applications and variants of the cuckoo search algorithm. *Applied Soft Computing*, 2017. Elsevier, v. 61, p. 1041–1059, 2017. Citado na página 13.

TORRESAN, C. et al. Forestry applications of uavs in europe: A review. *International Journal of Remote Sensing*, 2016. Taylor & Francis, v. 38, n. 8-10, p. 2427–2447, 2016. Citado na página 7.

TRASVIÑA-MORENO, C. A. et al. Unmanned aerial vehicle based wireless sensor network for marine-coastal environment monitoring. *Sensors*, 2017. Multidisciplinary Digital Publishing Institute, v. 17, n. 3, p. 460, 2017. Citado na página 5.

XU, Z.; NI, J. A study on key lpwan technologies. In: IOP PUBLISHING. *Journal of Physics: Conference Series*. [S.l.], 2021. v. 1871, n. 1, p. 012011. Citado na página 9.

YANG, X.-S. Bat algorithm for multi-objective optimisation. *International Journal of Bio-Inspired Computation*, 2011. Inderscience Publishers, v. 3, n. 5, p. 267–274, 2011. Citado 2 vezes nas páginas 19 e 20.

YANG, X.-S. Flower pollination algorithm for global optimization. In: SPRINGER. *International conference on unconventional computing and natural computation*. [S.l.], 2012. p. 240–249. Citado 2 vezes nas páginas 16 e 18.

YANG, X.-S. Nature-inspired optimization algorithms: Challenges and open problems. *Journal of Computational Science*, 2020. Elsevier, v. 46, p. 101104, 2020. Citado na página 13.

YANG, X.-S.; DEB, S. Cuckoo search via lévy flights. In: IEEE. *2009 World congress on nature & biologically inspired computing (NaBIC)*. [S.l.], 2009. p. 210–214. Citado 3 vezes nas páginas 13, 14 e 15.

YANG, X.-S.; HE, X. Bat algorithm: literature review and applications. *International Journal of Bio-inspired computation*, 2013. Inderscience Publishers Ltd, v. 5, n. 3, p. 141–149, 2013. Citado na página 19.

YANG, X.-S.; KARAMANOGLU, M.; HE, X. Flower pollination algorithm: a novel approach for multiobjective optimization. *Engineering optimization*, 2014. Taylor & Francis, v. 46, n. 9, p. 1222–1237, 2014. Citado na página 17.

YUHENG, Z.; LIYAN, Z.; CHUNPENG, L. 3-d deployment optimization of uavs based on particle swarm algorithm. In: IEEE. *2019 IEEE 19th International Conference on Communication Technology (ICCT)*. [S.l.], 2019. p. 954–957. Citado na página 6.

ZHICAI, R.; JIANG, B.; HONG, X. A cooperative search algorithm based on improved particle swarm optimization decision for uav swarm. In: IEEE. *2021 IEEE 6th International Conference on Computer and Communication Systems (ICCCS)*. [S.l.], 2021. p. 140–145. Citado na página 6.

ZHOU, Y.; RAO, B.; WANG, W. Uav swarm intelligence: Recent advances and future trends. *IEEE Access*, 2020. IEEE, v. 8, p. 183856–183878, 2020. Citado na página 6.

ZORBAS, D.; O'FLYNN, B. A network architecture for high volume data collection in agricultural applications. In: IEEE. *2019 15th International Conference on Distributed Computing in Sensor Systems (DCOSS)*. [S.l.], 2019. p. 578–583. Citado na página 5.