IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Date of publication xxxx 00, 0000, date of current version xxxx 00, 0000.

Digital Object Identifier 10.1109/ACCESS.2017.DOI

FA-CRAN: a Firefly Algorithm for dynamic BBU-RRH mapping in Cloud/Centralized Radio Access Networks

MARIANE DE PAULA DA SILVA GONÇALVES IMBIRIBA[2], ERMÍNIO AUGUSTO RAMOS DA PAIXÃO[2], ALBERT EINSTEIN COUTINHO DOS SANTOS[1], CARLOS ANDRÉ DE MATTOS TEIXEIRA[2], RAFAEL FOGAROLLI VIEIRA[2], DANIEL DA SILVA SOUZA[2], IGOR WENNER SILVA FALCÃO[2], DIEGO LISBOA CARDOSO[1, 2]

[1]Faculty of Computing and Technology, Federal University of Para, Belém, Pará, Brazil, CO 66075-110 BR
[2]Post-Graduate Program in Electrical Engineering, Federal University of Para, Belém, Pará, Brazil, CO 66075-110 BR

Corresponding author: Mariane de Paula da Silva Gonçalves Imbiriba (e-mail:goncalvesmariane@itec.ufpa.br).

This work was supported in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior-Brasil - CAPES under Finance 001, in part by the National Counsel of Technological and Scientific Development - CNPq, and in part by the Federal University of Para - UFPA, by the Dean of Research and Graduate Studies at UFPA.

ABSTRACT C-RAN (Cloud Radio Access Network) is an architecture designed to the new generation of mobile networks. It has handled many problems arising from the 4-generation network, and made several improvements, such as centralized processing and energy efficiency, among others. However, time-varying traffic, known as the tidal effect, impairs the network by making resource allocation less efficient, and this affects network performance in terms of problems related to blocked users and power consumption. This study seeks to evaluate an optimized mapping model between RRH (Remote Radio Head) and BBU (Base Band Unit) by providing a fairer and more efficient load balancing. In addition, this solution is compared with some key algorithms used in the literature for addressing optimization problems. The results demonstrated that, owing to its effective search feature, the Firefly algorithm, was the most promising system, since it obtained better performance measures than the others.

INDEX TERMS C-RAN; Balancing; Firefly Algorithm.

I. INTRODUCTION

According to Cisco's projections [1], by the end of 2028, there is expected to be 9.2 billion mobile devices, with an average consumption of 46 GB per month and a 26% growth in mobile data traffic. At the same time, Ericsson [26] predicts that five billion 5G subscriptions (55% of the total), will be activated, and ensure an extensive coverage of 85%. It is also expected that 5G networks will be responsible for 70% of mobile traffic, and thus handle all the current growth. This convergence of forecasts by Cisco and Ericsson envisages a distinct scenario for the evolution of mobile technology until 2028.

The exponential increase in connected devices and the growing demand for communication services necessitate a reevaluation of existing network architectures. The Distributed Radio Access Network (D-RAN), which traditionally has decentralized functions to base stations, faces challenges in meeting new requirements, since it incurs high capital costs and significant operational charges resulting from additional base stations [2]. In response to these drawbacks,, the Cloud Radio Access Network (C-RAN) architecture has emerged , and made far-reaching innovations. In C-RAN, the processing and control functions, which were traditionally performed at base stations, are centralized in what is called the Baseband Unit (BBU). This component is responsible for signal processing, resource management, and coordinating communication with Remote Radio Heads (RRH), which, in turn, are responsible for specific radio functions such as wireless signal transmission and reception. An approach that is adopted by centralizing processing in a cloud environment, offers benefits such as enhanced energy efficiency, better resource utilization, and increased flexibility when addressing the challenges of modern communication networks [3].

FIGURE 1. Traditional C-RAN architecture.

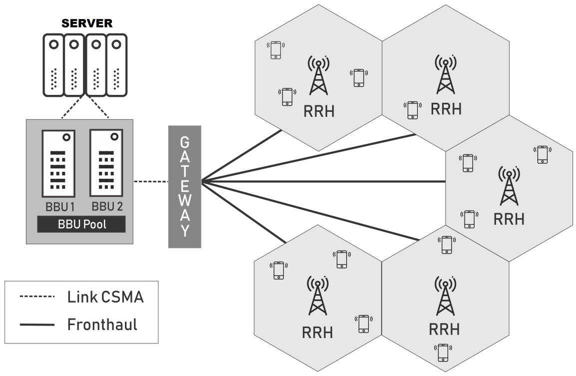


Figure 1 1 illustrates the C-RAN architecture, which has base stations that can be divided into three main components: RRH, fronthau / backhaul links, and BBU [6] [7]. The main features of these units are explained below.

- RRH: RRHs are made up of RF circuitry, such as up-and-down signal converters, ADC/DAC circuitry, and an optical interface to handle the physical-layer signals. The user equipment is connected to the antenna unit located at the RRH via a fronthaul or microwave link. The RRH increases coverage and reduces the C-RAN CAPEX (Capital Expenditure).
- Front and backhaul links: the connection between RRH and BBU is referred to as the fronthaul. A fronthaul link can be either wired or wireless. Fiber is more suitable for the fronthaul network because of its higher bandwidth and lower cost. The backhaul in C-RAN, on the other hand, connects the BBU and the main network for data and application control. The use of advanced features such as SDN and NFV in the backhaul enhances both the capacity and QoS efficiency of the C-RAN.
- BBU: the BBU is a centralized management and processing unit residing in a BBU pool. The BBU uses cloud computing and virtualization technologies to achieve a scalable sharing of physical resources and spectrum control. Each BBU represents a packet classifier and a hypervisor in a group of Virtual Boxes (VBs). The baseband processing is performed in a VB. The hypervisor manages all the VB executions carried out in the BBU [8].

C-RAN has the advantages of high capacity, wide coverage, low costs, high energy efficiency, and load balancing, and has been extensively implemented in the industry. In addition to the traditional spectrum resources (channel, bandwidth and power), C-RAN has new additional resources that can be orchestrated - the BBU resources. According to [9], the processing charge in each BBU is determined by the total number of simultaneous active users in the BBU sectors. However, there are both hardware and software restrictions, traditionally called Hard Capacity (HC), which are designed to define the limit of the number of active users in each sector of the BBU, and if they are poorly managed, it can lead to data and performance losses.

The variability in demand for Internet connection traffic, known as the tidal effect, can cause an underutilization of resources and deterioration of network performance, which represents one of the main challenges of the C-RAN architecture [6]. Poor radio and BBU resource orchestration can result in some sectors being overloaded, while others are underutilized, and make it impossible for new users to gain access to the network. Thus, appropriate resource management and configuration methods must be introduced to optimize C-RAN performance, while including a suitable response time and maximizing the number of users served.

Resource balancing and orchestration optimization is a high-dimensional, nonlinear, multi-constraint discrete optimization problem. Researchers have employed several methods to solve this problem, which can be roughly divided into three categories: an accurate algorithm, a heuristic algorithm based on priority rules, and an intelligent algorithm [1]- [3]. The results of the precise algorithm are accurate but very time-consuming, and are thus difficult to apply to large and complex problems; a heuristic algorithm based on optimization rules is fast but the quality of the optimization results depends on the merits of the rules and the universal application of the algorithm is poor. An intelligent algorithm is a kind of metaheuristic algorithm that simulates physical phenomena during its biological evolution in nature [4], [5]. It has global, parallel, efficient, and universal features, and is of often used to solve complex engineering optimization problems.

The aim of this study is to examine in greater depth the use of metaheuristics for resource orchestration in 5G networks and beyond, while taking into account aspects of performance and effectiveness. In this way, this paper seeks to make an evaluation of a self-organizing C-RAN architecture with the ability to reconfigure its resources, even in varying traffic conditions, as well as to improve network efficiency through a Key Performance Indicator (KPI). A controller residing within the BBU pool is used to monitor load conditions and determine the appropriate RRH-BBU configuration in the network. This problem is handled by the bio-inspired Firefly Algorithm (FA), an algorithm proposed by [13] which belongs to the group of stochastic algorithms, as it performs its search for solutions at random. Their search method is based on the intensity of the light that the fireflies emit. [13]. Other bio-inspired techniques were used for benchmarking purposes to determine the best algorithm that can carry out this optimization task. One is based on Particle Swarm Optimization (PSO), which is designed and implemented for this end [8][11] and another on the Bat Algorithm (BAT). A more recently created algorithm was used for an equivalent purpose [12], and both algorithms have been widely used in the literature to solve optimization problems such as resource allocation.

It is important to make a proper choice of the metaheuristic that will be used , as the performance of these algorithms when tackling each type of problem must be noted. This is because their performance may vary, depending on the particular features of the load balancing problem, the extent of the problem and the parameters of the algorithm. In light of this , this study concentrates on conducting experiments and making a comparison of the performance of different algorithms when addressing a given RRH- BBU resource and load-balancing orchestration problem

The research contributions of this article are as follows:

- Designing an efficient mapping model between RRH and BBU to optimize load balancing in the C-RAN network, while taking account of the tidal effect and its implications for resource allocation, with an emphasis on equity, efficiency, and the reduction of blocked users. The purpose of this is to improve network performance globally, through an additional focus on energy efficiency.
- Making a detailed comparison of the proposed system with the main algorithms used in optimization problems in the literature. This involves highlighting the effectiveness of the Firefly algorithm in seeking promising solutions and achieving a better performance, as well as emphasizing its capacity optimization .
- Assisting in the advancement of mobile networks, especially by transitioning to the next generation, gaining insights and finding appropriate solutions for advanced networks like 5G and future technologies.

This article is structured as follows: Section II discusses the literature with regard to the main features of the architecture and the problems addressed in this paper. Section 3 describes the Firefly Algorithm. Section 4 examines the architecture and simulation methodology employed in this study . The results are shown in Section 5, followed by the conclusion in Section 6.

II. RELATED WORK

Optimization problems are common in many areas of real-world applications. In this context, several bio-inspired techniques based on swarm intelligence have been applied in several areas such as power systems, vehicle routing, robot planning, and telecommunications. This especially applies to the concern about the current and upcoming increase in traffic flow in the next few years, particularly since the C-RAN architecture was designed to mitigate the problem of these demands. This architecture has dynamic features , such as self-organization and an ability to intelligently redistribute resources in different scenarios.

It was recommended that a resource manager should be located inside the BBU to monitor the traffic conditions and adjust the BBU-RRH mapping. A bio-inspired AB algorithm was used to optimize this process. In addition, two algorithms from the literature were implemented for benchmarking purposes. The results showed that the AB algorithm proved to be more efficient and had a shorter execution time.

The authors of [11], also highlight the C-RAN architecture as a possible alternative, owing to its processing power and ability to reconfigure and map users in an intelligent way. A method to optimize the quality of service (QoS) of the architecture was investigated by using a KPI to reduce the number of blocked UEs. The PSO algorithm was used to this end to match users with the KPI and optimize the QoS. TThe authors obtained satisfactory results, which in fact proved to be potentially better than the results in the literature.

The study carried out in [13] underlines the effectiveness of bio-inspired algorithms in the field of optimization. It also introduces a new, carefully designed and well explained technique, referred to as Firefly Algorithm (FA). The authors compare the effectiveness of the FA with other consolidated metaheuristic approaches such as Particle Swarm Optimization (PSO). The new algorithm was used to solve a series of optimization problems with several functions. The results show that the FA outperformed both algorithms when they were compared, which proved that it might be more powerful in solving these problems and more suitable for dealing with future problems. An attempt will be made to further validate and embed this new method in future research.

In seeking to optimize the mapping between the BBUs and RRHs, the authors of [14] applied the Particle Swarm Optimization (PSO) algorithm, which belongs to the area of Swarm Intelligence, to ensure intelligent utilization of C-RAN resources. The key objective of this study is to reduce network costs through dynamic resource allocation, while also seeking to reduce the energy consumption of C-RAN resources, which directly affects the allocation of RRHs. The results showed a significant reduction in energy consumption, together with a maximum conservation of energy efficiency, which led to research aimed at more sustainable and economically viable communication networks.

In [17] an energy-efficient C-RAN is proposed for the 5G network. To achieve this, the authors use a Particle Swarm Optimization (PSO) algorithm for resource allocation in C-RAN. The main objective of the study is to dynamically allocate BBU resources to RRHs that are based on traffic and reduce the energy consumption of centralized BBU resources. The proposed PSO algorithm achieves a 90% reduction in energy consumption and maximizes energy efficiency, when compared with the existing works in the literature.

The author in [18] proposes a solution for load balancing in 5G C-RAN based on BBU-RRH dynamic mapping that supports IoT communications. For this, the Markov model is employed to predict the traffic load based on the current location of users for each cell. The Ant Colony Optimization (ACO) technique is employed to find the optimized mapping between BBU and RRH. The proposed solution reconfigures BBU and RRH logical connections based on IoT network traffic prediction, resulting in a balanced load across the IoT network. The proposed solution aims to minimize the number of blocked connections in the IoT network to the lowest possible value, leading to maximizing QoS.

In [10], finds a solution for load balancing in 5G C-RAN which is based on the BBU-RRH dynamic mapping that supports IoT communications. The Markov model is employed for this to predict the traffic load that is based on the current location of users for each cell. The Ant Colony Optimization (ACO) technique is employed to find the optimized mapping between BBU and RRH. The proposed solution reconfigures BBU and RRH logical connections on the basis of IoT network traffic prediction, and results in a balanced load across the IoT network. The aim of this solution is to reduce the number of blocked connections in the IoT network to the lowest possible value, as a means of maximizing the QoS.

In [17] there is a discussion about the two-level coverage. In the second stage, two algorithms are used to reduce the number of BBU servers, and hence save energy, which is carried out by consolidating BBUs. Both proposed solutions achieve good results, and thus can ensure energy savings without losing QoS quality.

The architectures proposed in [24] and [25] demonstrate their effectiveness in the comprehensive optimization of radio frequency, optical spectrum, and BBU processing resources, and seek to maximize radio coverage and meet Quality of Service (QoS) requirements. The works show promising results with regard to these essential factors . However, it should be noted that they do not delve into the orchestration of resources in BBU pools in detail. In particular, there is a failure to fully address the question of the effects of the tidal effect, which can result in user blocking caused by an inadequate distribution of resources. This gap in the analysis could have an adverse effect on operational efficiency in dynamic scenarios where traffic demand varies.

In [19], energy efficiency of the system through the allocation of resources and power control. The authors also reduced energy consumption by turning on/off RRHs on the basis of the current user distribution. The problem was modeled with an energy efficiency (EE) maximization that was constrained to provide full frequency reuse across RRHs. The solution breaks the problem down into several stages to reduce the complexity. First, the RRHs that must be turned on/off are selected and then based on the operational RRHs; the assignment of users and the transmission power of the RRHs are optimized. Finally, any unattended user is assigned to the macro base station. A performance review shows that the solution significantly improves the EE system and energy consumption, when compared with other solutions in the literature and achieves an improvement in EE of more than 57%.

III. RRH-BBU MAPPING AND RELATED KPIS

Resource orchestration or RRH-BBU mapping involves determining what is suitable for making logical connections between BBU, RRH, and sectors where there is load balancing . This improves the network performance function at period $t + 1$ under a given network condition at period t that is based on selected ll KPIs [16]. Thus it is necessary to know the position and throughput demand of the user in the period t. This section describes the essential features needed to orchestrate radio and BBU resources for 5G and beyond networks, including the RRH-UE allocation process, propagation modeling, and the KPIs used for load balancing across BBU sectors.

A. SYSTEM MODEL

The Stanford University Interim model (SUI) was used to calculate the propagation loss in metropolitan environments [20]. Equation 1 presents a model of the SINR (Signal to Interference plus Noise Ratio) relationship for a specific user k connected to an RRH (n):

$$SINR_k = \frac{P_r(k)}{\alpha^2 + I_k} \tag{1}$$

Where $P_r(k)$ is the power received by the user k, α^2 is the thermal noise power, and I_k is the intercellular interference of RRH. The power received by the user can be calculated by Equation 2 that links the transmission power Pot with the propagation loss obtained by the SUI-Type-A model.

$$P_r k = \frac{10^{Pot}}{\alpha^2 + I_k} \tag{2}$$

The three values that make up L_{SUI} can be calculated by means of the following three equations:

$$L_{SUI} = A + 10\gamma log\frac{d}{d_o} + SE, d > d_o, \tag{3}$$

$$A = 20log\frac{4\pi d_o}{\lambda}, \tag{4}$$

$$\gamma = a - bh_b + \frac{c}{h_b} \tag{5}$$

The data rate of each user is estimated using the Shannon capacity expressed in the Equation 6, with B being the system bandwidth [20].

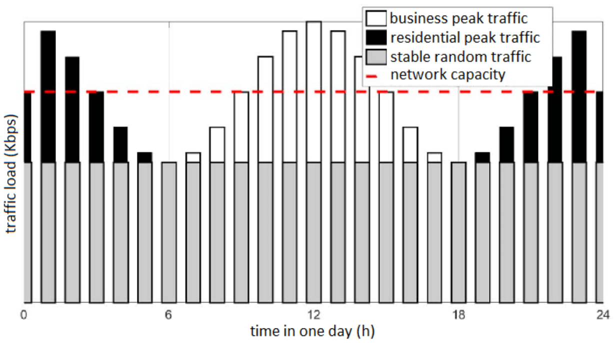
$$C_k = B * log2(1 + SINR_k) \tag{6}$$

The tidal effect, known as the influence by predictable patterns of human movement with mobile devices, impacts in the network pattern [21].

Figure 2 highlights this phenomenon, by showing the traffic load measurement calculated by the article [6] in the districts of New York. This dynamic requires an in-depth investigation on account of the crucial information it provides for areas such as traffic engineering, network design, load balancing, and pricing, as is proved by the ievidence in the study [6]. The measurement of traffic loads, expressed in kilobits per second (kbps) throughout the day, relies on monitoring tools to collect data on the volume of data transmitted and received at regular intervals, usually every hour. The calculation involves adding up the total volume of data in bits during each interval and converting the sum total to kbps, which is a means of gaining a valuable insight into daily usage patterns,traffic peaks, and temporal variations. This process facilitates the optimization of network capacity and efficient resource planning.

This article has been accepted for publication in IEEE Access. This is the author's version which has not been fully edited and
content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2023.3347341

FIGURE 2. Traffic Flow Patterns per hour in New York city



B. KPI FOR BBU BALANCING

To optimize RRH-BBU mapping and analyze its effectiveness in a dynamic traffic network, an objective function (Equation 7) was used as a KPI with the aim of distributing UEs as uniformly as possible between the sectors of each BBU; this took account of the maximum number of UEs that each sector can support. The KPI was based on [16].

$$U_{s(i)} = \sum_{j=1}^{N} C_j R_s, S = 1, 2, .., K \tag{7}$$

Where:
$U_{s(i)}$ is the number of UEs in the sector;
N is the total PRRHs;
K is the total sectors;
C is the number of UE connected in PRRHj;
R is a binary variable where it takes the value 1 if PRRHj is allocated to sector s.

The vector Us used in the objective function will vary in accordance with the number of UEs in the network to ensure that the sectors are balanced, and hence their respective BBUs are balanced as well. All the possible Us for all the K (sectors) will be tested to obtain the smallest possible value (KPImin).

The equation proposed in [14] wwas used as a restrictive measure. This process involves reducing the number of blocked UEs and, hence, maximizing QoS, as shown in Equation 8.

$$0, if(UsHC) < 0or(UsHC)if(UsHC) >= 0 \tag{8}$$

The output of the model is given by the following vector: $Si + 1j = Si + 11, Si + 12, ..., Si + 1N$, which represents the sectors of the BBUs Si+1j and PRRHs that have been allocated to these sectors.

IV. IMPLEMENTATION OF FIREFLY ALGORITHM FOR RRH-BBU BALANCE OPTIMIZATION

In recent decades, several algorithms have emerged that are inspired by natural phenomena , especially in the field of metaheuristic numerical optimization. Techniques such as those employed by evolutionary algorithms and swarm algorithms, are often based on populations. The former comprise computational techniques based on biological evolutionary principles such as natural selection, mutation, and genetic inheritance. Swarm algorithms, on the other hand, correspond to systems that seek to replicate the swarm intelligence that is generally observed in nature, in which a group of agents cooperates collectively to achieve some objective [13], [15].

The Firefly Algorithm stands out among bio-inspired algorithms as a swarm intelligence technique based on the bioluminescence of fireflies. According to the authors of [13], some properties of fireflies were simplified to develop FA, such as the following: (1) the fact that all fireflies are attracted to one another regardless of gender; (2) the capacity for attraction capacity is proportional to their brightness, and diminishes as decreasing with the distance between them widens; (3) if there was no firefly brighter than the others, they all moved randomly. Thus, it is understood that the brightness of a firefly can be directly affected by the medium or by the objective function of the problem in question. In a maximization problem, for example, the brightness can be proportional to the value of the objective function. Taking these characteristics into account and based on the three previous rules, the operation of the algorithm is summarized by Algorithm 1.

Algorithm 1: Firefly Algorithm pseudocode.

```
Data: Objective Function f(x), x = (x_1, ..., x_d)^T;
      Generate initial firefly population x_i
      (i = 1, 2, ..., n); Sets light intensity I_i in x_i
      using f(x_i); Sets absorption coefficient γ
Result: Best solution found
1  σ(S) = 0;
2  while t < MaxGeneration do
3      for i = 1 to n do
4          for j = 1 to n do
5              if I_j > I_i then
6                  Move the firefly i towards the firefly j;
7              end
8              Calculate β(r) = β_0 e^{γr^2};
9              Evaluate new solutions and update light
                intensities;
10         end
11     end
12     Rank the fireflies and find the best one;
13 end
```

Initially, the population of fireflies is generated at random.. The light intensity values, which correspond to the value of the objective function, are then calculated and assigned to each firefly. The absorption coefficient of the medium is defined in line 4. If the light intensity of firefly i is less than that of j, i is attracted by j (lines 10-12). Then, on line 13, the attraction value of a firefly is calculated by means of

This article has been accepted for publication in IEEE Access. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2023.3347341

the equation $\beta(r) = \beta_0 e^{\gamma r^2}$, in which r corresponds to the Euclidean distance between fireflies i and j, β_0 the initial attractiveness at r_0 and γ the absorption coefficient of the medium. The generated solutions are evaluated on line 16 after their respective light intensities have been calculated. Finally, after the fireflies have been ranked in terms of their brightness, the one with the highest luminous intensity is chosen as the best solution.

V. ARCHITECTURE AND SIMULATION

In the proposed method, the BBU pool host manager uses a bioinspired algorithm to perform the RRH-BBU mapping by taking note of the performance indicator recommended in [12], through which it obtained acceptable balancing metrics for its users. The use of hardware-embedded metaheuristics has already been widely discussed in the literature in various branches of engineering, and further references/discussion on this point can be found in [22] [23]. In this context, a simulation is carried out that is designed to optimize the mapping process and evaluate the best bioinspired algorithm in the network that can perform resource balancing. However, this paper intends to conduct a performance analysis to investigate which will outclass [11] as a means of mitigating this problem, and find a good solution with a shorter convergence time, that will improve the QoS of real-time flows in a context of high user mobility.

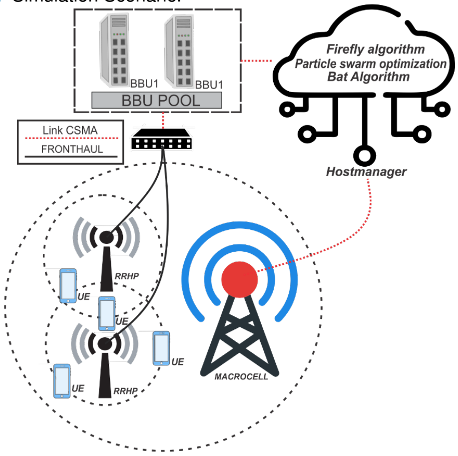
The scenario consists of 19 RRHs that are randomly dis- tributed between two BBUs divided into three sectors , which are managed by a BBU pool. The RRHs serve a total number of 1500 users. The prospect of reducing the number of blocked UEs is evaluated after the RRHs have been distributed and allocated to their respective BBUs, since an unbalanced network overloads certain sectors, thus leading them to overheat and perhaps cause performance losses. The simulations were performed by means of the Matlab1 simulator because of its extensive documentation, and the fact that it makes it easier to create new scenarios or adjust them.

The scene is made up of the H-C-RAN architecture shown in Figure 3. RRHs, UEs, Macro, and BBUs are located in the New York area and are interconnected by optical fiber; the BBUs are grouped in the BBU pool and managed by the host manager. The design of these devices allows algorithms to be implemented that can optimize the resource reconfiguration process. After the RRHs have been distributed and allocated to their respective BBUs, a decision is made about whether or not more BBU settors are necessary. Owing to their extensive documentation, (which makes it easier to create and adjust scenarios), the simulations were carried out by means of Matlab.

As shown in Figure 3, the infrastructure consists of BBU processors, RRH Macrocells, and user-facing devices. The host manager is responsible for implementing the network functions that are a) tbased on a physical infrastructure, b) in infrastructure cabinets and c) running bio- inspired algorithms. In our scenario, the host manager activates the hourly algorithms to ensure the network is more effectively balanced.

The three algorithms followed the steps outlined in Figure 3.

FIGURE 3. Simulation Scenario.



VI. SIMULATION AND ANALYSIS

This section outlines a performance evaluation of the proposed optimization for load balancing and resource orchestration with the aid of the Firefly Algorithm (FA). The evaluation t takes account of three essential metrics: execution time, number of iterations, and standard deviation. Two widely recognized algorithms with regard to this type of problem [12] and [8] were chosen as benchmarks for purposes of comparison. The first, Particle Swarm Optimization (PSO), is grounded on the social behavior of birds and shoals of fish, where a population of candidate solutions, called particles, collaborates and moves about in the search space, influenced by the optimal positions that can be found both individually and globally. The second benchmark, Bat Algorithm (BAT), emulates the habits of bats when foraging for food, and adopts manoeuvres such as echolocation and adaptive movement to adjust candidate solutions in the search space. The results of the balancing technique were calculated by a machine equipped with an Intel(R) Core (TM) i5-3317u processor, clocked at 1.7GHz, and with 8GB of DDR3 RAM. A population of 30 particles was generated, and the stopping criterion was set at 100 iterations for each algorithm. The rest of the parameters can be found in Table 1.

Given the inherent stochastic nature of the algorithms, it is essential to subject the techniques to multiple iterations and employ descriptive statistical metrics to obtain a more reliable analysis of the results. In the context of this study, 30 executions were carried out, by employing median and deviation from the norm, as metrics to calculate the measure of dispersion in the generated values.

Furthermore, we incorporated three distinct scenarios in the simulation, each featuring varying numbers of users and BBUs. The purpose of this deliberate variation was to provide

TABLE 1. Simulation Parameters

Parameters	Values
Propagation loss (*MRRH*)	COST231
Propagation loss (*PRRH*)	SUI-TYPE A
Transmit Power (*MRRH*)	43 dBm
Transmit Power (*PRRH*)	23 dBm
Total Scene Area	4km²
PRRH Height	16m
Coverage area PRRH	150m
Coverage area MRRH	4km
Confidence Interval	95%
Number of Experiments	31

TABLE 2. Results of the Bioinspired Algorithms 500.

Evaluated Metrics	Firefly	P.S.O.	B.A.T.
Mean Execution Time	**1,23 ms**	7,36 ms	6,1 ms
Execution Time Std. Deviation	**0,23**	0,39	0,36
Mean Number of Iteractions	**3ª**	11ª	10ª
Number of Iteractions Std. Deviation	**0,17**	0,32	0,30

TABLE 3. Results of the Bioinspired Algorithms 1000.

Evaluated Metrics	Firefly	P.S.O.	B.A.T.
Mean Execution Time	**2,19 ms**	10,13 ms	9,80 ms
Execution Time Std. Deviation	**0.34**	0.56	0.48
Mean Number of Iteractions	**10ª**	36ª	31ª
Number of Iteractions Std. Deviation	**0,30**	0,46	0,42

TABLE 4. Results of the Bioinspired Algorithms 1500.

Evaluated Metrics	Firefly	P.S.O.	B.A.T.
Mean Execution Time	**3,91 ms**	18.25 ms	16,34 ms
Execution Time Std. Deviation	**0.45**	0.81	0.78
Mean Number of Iteractions	**17ª**	42ª	37ª
Number of Iteractions Std. Deviation	**0,36**	0,48	0,46

a more comprehensive understanding of the effectiveness of each technique.

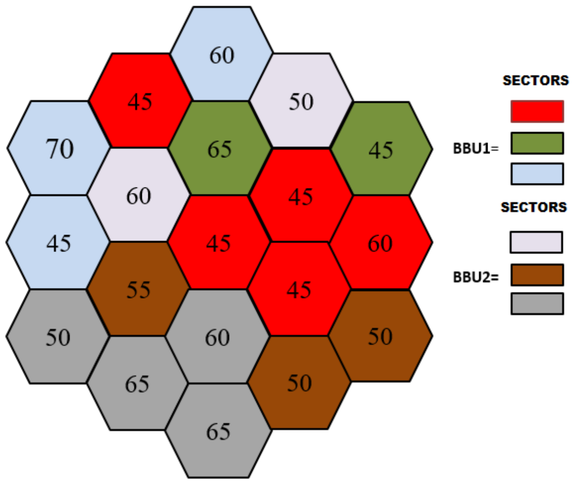
Tables 2, 3, and 4 display the average execution time and standard deviation of the number of iterations required to achieve convergence in scenarios with 500, 1000, and 1500 users, respectively.

The data in Table 2 confirm this effectiveness, by showing that, on average, FA had an approximately 83.29% faster execution time than PSO and was 79.84% faster than BAT. Furthermore, in terms of iterations, FA reached the desired solution approximately 72.73% faster than PSO and about 70% than BAT. Table 3 further underlines the remarkable efficiency of FA. With regard to its average execution time, the algorithm had an acceleration rate of approximately 78.38% compared with PSO and an improvement in speed of 77.65% when compared with BAT. As regards the iteration count, FA reached the expected solution about 72.22% times faster than PSO and was approximately 67.74% faster than BAT.

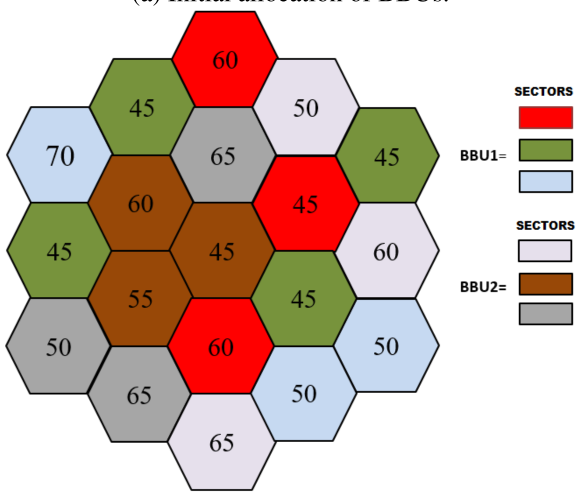
Following the same line of analysis, Table 4 corroborates the consistent effectiveness of the FA algorithm. Concerning the average execution time, the FA performed approximately 78.58% faster than PSO and 76.07% than BAT. In terms of iteration count, the FA reached the expected solution about 59.52% times faster than PSO and approximately 54.05% faster than BAT.

An analysis of the three displayed tables, consistently shows the clear superiority of FA over traditional methods . In Tables 2, 3, and 4, FA demonstrated exceptional efficiency by significantly outperforming PSO and BAT in average execution time and iteration counts.

In assessing the algorithm's robustness, both the population and the sample space were expanded to test the performance when stretched to the limit. Even with this notable increase, the FA algorithm maintained a superior performance to that of its competitors.



(a) Initial allocation of BBUs.



(b) Final allocation of BBUs.

FIGURE 4. 19 RRHs governed by 2 BBUs in the BBU pool, each BBU covers 3 sectors.

Figure 4 illustrates the practical results of this performance. From an observation of the hexagons representing the Remote Radio Heads (RRHs) and the allocation of blocked User Devices (UDs), the visual display in (a) without the balancing process highlights that there are network losses. In contrast, with the algorithm's execution, b) displays a fairer, more balanced, and optimized distribution. This process results in the reduction of UDs, and confirms the effectiveness of the balancing when it is facilitated by the FA algorithm in network optimization.

The graph that displays a convergence curve of the evaluated algorithms is shown in 567. In 5, when 500 UEs were added, the FA converged 83% faster than PSO, compared with the 79.8% of BAT, respectively. Conversely, in 6, the FA outperforms the others while maintaining a high efficiency ratio, with a difference of 78.3% compared with PSO and 77.6% compared with BAT. Finally, in 7, where the number of UEs is tripled, the FA obtains the best solution with just half as many interactions as the others, and kept an efficiency rate that is at least 70% higher than that of the others.

It should be remembered that a a performance comparison between different optimization algorithms often depends on the nature of the problem. An algorithm that performs well in one problem may not be as effective in another. Thus , the context and features of an optimization problem must be carefully taken into account when selecting an algorithm . In the context of load balancing optimization, the FA can achieve faster convergence than PSO or BAT. This can be attributed to several distinctive features: a) Attraction Mechanism: The FA employs a mechanism where fireflies are attracted to brighter fireflies, and thus represent better solutions. This mechanism helps the algorithm adapt quickly to promising solutions, and assist load balancing. b) Intensity-Based Communication: Fireflies communicate on the basis of their light intensities, and this enables them to share load information and adapt to positions more effectively. c) Adaptive Behavior: Fireflies in the FA behave in an adaptive manner by adjusting their powers of attraction and movement to the local and global environment. This adaptability allows them to have a dynamic response to changes in load conditions. d) Exploration-Exploitation Tradeoff: like other metaheuristic algorithms, the FA seeks to balance the exploration of the search space (i.e. attempt to find new solutions) with an exploitation of known solutions (i.e. by optimizing existing solutions). In load balancing optimization, this balance is crucial to ensure that the search space can be efficiently explored to find the best load distribution solutions while continuing to optimize measures that that can clearly maintain a good balance.

The 8 shows how these three tactics are planned. Remarkably, the methodology employed used in the published literature results in sectors that are free of blocked calls, although with a significantly imbalanced user distribution. On the other hand, the suggested approach demonstrated that there could be a more equal allocation of sectors and led to zero blocked calls.

Our statement explains that the adaptability of the algorithms allows them to dynamically respond to changes in load conditions by effectively balancing the load and making adjustments to position and intensity. More specifically, the Firefly Algorithm (FA) establishes a trade-off between exploration (finding new solutions) and exploitation (optimizing existing solutions) in the context of load balancing opti-

FIGURE 5. Convergence curve for 500 UEs.

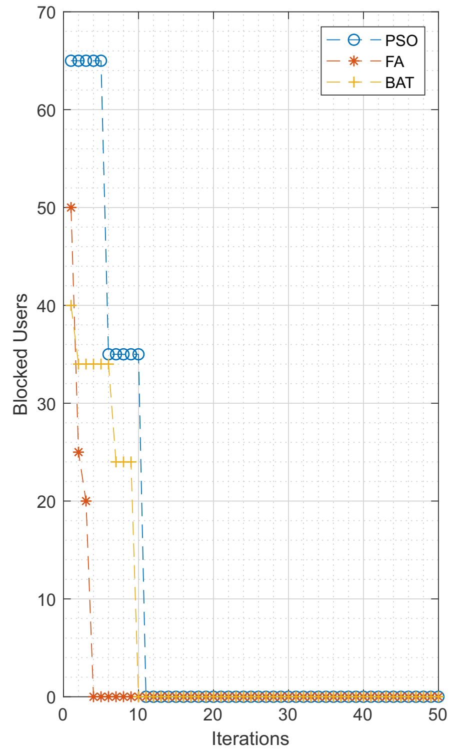


FIGURE 6. Convergence curve for 1000 UEs.

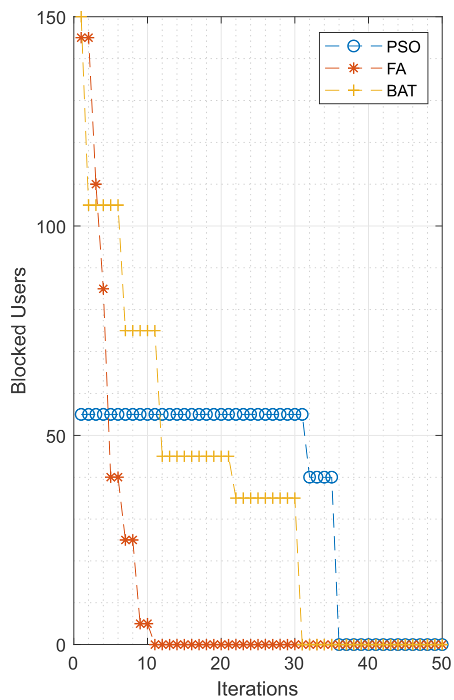


FIGURE 7. Convergence curve for 1500 UEs.

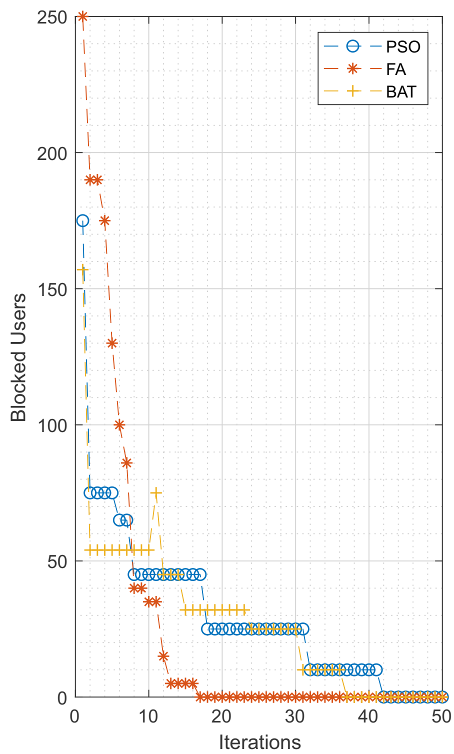
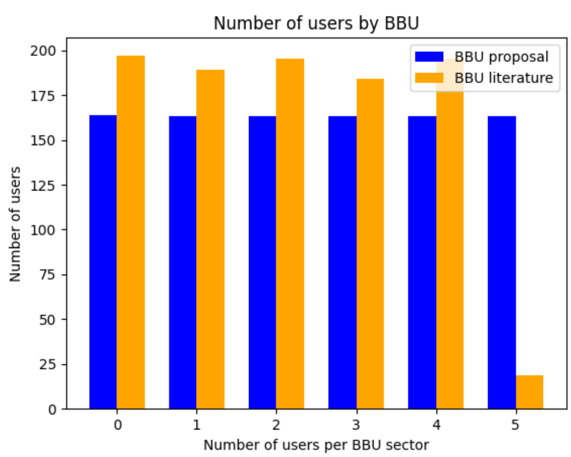


FIGURE 8. Convergence curve for 500 UEs.



mization. This balance is essential for efficiently exploring the search space while making use of previously identified solutions that have a good balance.

It should also be emphasized that the performance of these algorithms can vary, depending on the particular features they have such as the nature of the load balancing problem, the extent of the problem, and the parameters of the algorithm. A recognition of this underscores the importance of taking account of the specific context and requirements of the load balancing scenario when evaluating the performance of algorithms.

Furthermore, the following factors should be noted with regard to the complexity of the algorithms, . In each iteration, the Firefly algorithm updates the position and velocity of n populations of the created fireflies and then determines the best population. Thus, the complexity of the algorithm is $O(M * n)$, where M represents the maximum number of iterations and n represents the number of created populations. On the other hand, the PSO algorithm has a complexity of $O(M * n * nBBU + nBBU)$, where $nBBU$ represents the number of sectors in the current BBU. The first part, $M * n * nBBU$, represents the update d positions and velocities of the particles, while $nBBU$ represents the assignment of values for the best swarm found. Finally, the BAT algorithm has a complexity of $O(M * n * nBBU)$, which corresponds to the update of the position and velocities of the bats that make up the population. In light of this , it is clear that the Firefly algorithm has a better scalability and performance.

VII. CONCLUSION

The focal point of this study is on C-RAN networks, which have an architecture that centers on optimizing the mapping between RRH (Remote Radio Head) and BBU (Baseband Unit). Bio-inspired algorithms that are widely used in the literature were created to act as benchmarks with the aim of enhancing the mapping and load balancing of data in the network. Thus, it was possible to assess which of them performs better for this task. After analyzing the results, it was found that the Firefly Algorithm (FA) proved to be superior to the others in terms of efficiency and resource allocation. In view of this, future research could focus on quantifying and analyzing the specific benefits of these algorithms in reducing energy consumption, as this would make a valuable contribution to the sustainable and efficient development of mobile communication networks.

A. COPYRIGHT FORM

Authors must submit an electronic IEEE Copyright Form (eCF) upon submitting their final manuscript files. You can access the eCF system through your manuscript submission system or through the Author Gateway. You are responsible for obtaining any necessary approvals and/or security clearances. For additional information on intellectual property rights, visit the IEEE Intellectual Property Rights department web page at http://www.ieee.org/publications_standards/publications/rights/index.html.

REFERENCES

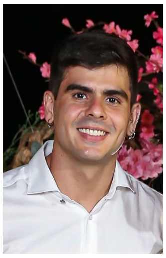
[1] "Ericsson mobility report," Ericsson.com, Nov. 2022.
[2] L. Wang et al., "Statistical Multiplexing Analysis with Quantized Computing Resource for Practical C-RAN," in ICC 2019-2019 IEEE International Conference on Communications (ICC), pp. 1-6, IEEE, 2019.
[3] Ericsson, "Ericsson's fiber fronthaul solution deployed for China Mobile's LTE C-RAN," http://www.ericsson.com/news/140707-ericssons-fiber-fronthaul-solutiondeployed_244099436_c, Nov. 2017.
[4] Y. Lin, L. Shao, Z. Zhu, Q. Wang, and R. K. Sabhikhi, "Wireless network cloud: architecture and system requirements," IBM J. Res. Dev., vol. 54, no. 1, pp. 4-1, 2010.
[5] Y. Zhang, L. Budzisz, M. Meo, A. Conte, I. Haratcherev, G. Koutitas, L. Tassiulas, M. Ajmone Marsan, and S. Lambert, "An overview of energy-efficient base station management techniques," in: Digital Communications-Green ICT (TIWDC), 24th Tyrrhenian International Workshop on, pp. 1-6, 2013.
[6] K. Chen and R. Duan, "C-RAN: The Road Towards Green RAN," China Mob. Res. Inst., vol. 2, 2011.
[7] M. Makhanbet, X. Zhang, H. Gao, and H. A. Suraweera, "An overview of cloud RAN: architecture, issues and future directions," in: International Conference on Emerging Trends in Electrical, Electronic and Communications Engineering, Springer, pp. 44-60, 2016.
[8] D. Mishra, P.C. Amogh, A. Ramamurthy, A. A. Franklin, and B. R. Tamma, "Load-aware dynamic RRH assignment in cloud radio access networks," in: Wireless Communications and Networking Conference (WCNC), pp. 1-6, 2016.
[9] N. Chen et al.,"Self-organizing scheme based on NFV and SDN architecture for future heterogeneous network", Mobile Networks and Applications, vol. 20, no. 4, pp. 466-472, 2015.
[10] M. Mouawad, Z. Dziong, and K. Addali, "RRH selection and load balancing through Dynamic BBU-RRH Mapping in C-RAN" in: 2019 IEEE Canadian Conference of Electrical and Computer Engineering (CCECE), pp. 1-5, IEEE, 2019.
[11] E. A. Ramos da Paixão, R. F. Vieira, W. V. Araújo and D. L. Cardoso, "Optimized load balancing by dynamic BBU-RRH mapping in C-RAN architecture," 2018 Third International Conference on Fog and Mobile Edge Computing (FMEC), Barcelona, Spain, 2018, pp. 100-104, doi: 10.1109/FMEC.2018.8364051.
[12] T. Yuvaraj, K. Ravi, and K. Devabalaji, "DSTATCOM allocation in distribution networks considering load variations using bat algorithm," Ain Shams Engineering Journal, vol. 8, no. 3, pp. 391-403, 2017.
[13] Xin-She. Yang, "Firefly algorithms for multimodal optimization," in: International symposium on stochastic algorithms, Springer, Berlin, Heidelberg, pp. 169-178, 2009.
[14] P. R. Adiraju and V. S. Rao, "Dynamically Energy-Efficient Resource Allocation in 5G C-RAN Using Intelligence Algorithm," EMITTER International Journal of Engineering Technology, pp. 217-230, 2022.
[15] Xin-She. Yang, "A New Metaheuristic Bat-Inspired Algorithm," Springer, Berlin Heidelberg, pp. 65-74, 2010.
[16] M. Khan, Z. H. Fakhri, and H. S. Al-Raweshidy, "Semistatic cell differentiation and integration with dynamic BBU-RRH mapping in cloud radio access network," IEEE Transactions on Network and Service Management, vol. 15, no. 1, pp. 289-303, 2017.
[17] ADIRAJU, Prasanth Rao; RAO, Voore Subba. Dynamically Energy-Efficient Resource Allocation in 5G C-RAN Using Intelligence Algorithm. EMITTER International Journal of Engineering Technology, p. 217-230, 2022.
[18] MOUAWAD, Mostafa; DZIONG, Zbigniew; EL-ASHMAWY, Ahmed. Load balancing in 5G C-RAN based on dynamic BBU-RRH mapping supporting IoT communications. In: 2018 IEEE Global Conference on Internet of Things (GCIoT). IEEE, 2018. p. 1-6.
[19] A. Srivastava, M. S. Gupta, and G. Kaur, "Energy efficient transmission trends towards future green cognitive radio networks (5G): Progress, taxonomy and open challenges," Journal of Network and Computer Applications, vol. 168, p. 102760, 2020.
[20] L. S. B. Castro, et al."COST231-Hata and SUI Models performance using a LMS tuning algorithm on 5.8 GHz in Amazon Region cities."Proceedings of the Fourth European Conference on Antennas and Propagation. IEEE, pp. 1-3. 2010
[21] I. W. S. Falcão et al., "The Heuristic for Hardware Dimensioning Considering Tidal Effect," Journal of Communication and Information Systems, vol. 35, no. 1, pp. 311-319, 2020.
[22] A. Ortiz et al., "Hardware implementation of metaheuristics through LabVIEW FPGA," Applied Soft Computing, vol. 113, p. 107908, 2021.
[23] H. Chen et al., "SEFSD: an effective deployment algorithm for fog computing systems," Journal of Cloud Computing, vol. 12, no. 1, pp. 1-15, 2023.
[24] H. Yang, J. Zhang, Y. Ji and Y. Lee, "C-RoFN: multi-stratum resources optimization for cloud-based radio over optical fiber networks," in IEEE Communications Magazine, vol. 54, no. 8, pp. 118-125, August 2016, doi: 10.1109/MCOM.2016.7537186.
[25] H. Yang, Q. Yao, B. Bao, A. Yu, J. Zhang and A. V. Vasilakos, "Multi-Associated Parameters Aggregation-Based Routing and Resources Allocation in Multi-Core Elastic Optical Networks," in IEEE/ACM Transactions on Networking, vol. 30, no. 5, pp. 2145-2157, Oct. 2022, doi: 10.1109/TNET.2022.3164869.
[26] Ericsson, "Ericsson Mobility Report: Global 5G Growth Amid Macroeconomic Challenges,"2022,https://www.ericsson.com/en/press-releases/2022/11/ericsson-mobility-report-global-5g-growth-amid-macroeconomic-challenges, Acessado em 11 de novembro de 2023.

MARIANE DE PAULA DA SILVA GONÇALVES IMBIRIBA is a doctoral student with an emphasis on applied computing and Research in Networks and Distributed Systems, in the Graduate Program in Electrical Engineering (PPGEE) at the Federal University of Pará. She works at the Operational Research Laboratory and is dedicated to the area of research on new architectures for 5G, load balancing, resource allocation, applied computational intelligence and optimization techniques.

ERMÍNIO AUGUSTO RAMOS DA PAIXÃO Graduated in Computer Networks at the University of Amazonia (2015), Master's Degree in Electrical Engineering at the Federal University of Pará (2018), PhD candidate in Electrical Engineering at the Federal University of Pará (2019), and member of the Operational Research Laboratory, where he works in the area of high performance networks, 5G, QoS and computational intelligence, with an emphasis on optimization techniques.

ALBERT EINSTEIN COUTINHO DOS SANTOS Undergraduate in Computer Engineering at the Federal University of Pará (UFPA). He was a fellow of the Laboratory of Didactic Innovation in Physics (LIDF), where he worked on Arduino prototyping and IoT. [give dates] He is interested in Data Science Programming and Computational Intelligence. He is currently the holder of a scientific initiation scholarship which was awarded by the Institutional Program for Scientific Initiation Scholarships (PIBIC) at the Operational Research Laboratory (LPO) of the Federal University of Pará.

CARLOS ANDRÉ DE MATTOS TEIXEIRA is a PhD candidate in Machine Learning at the Federal University of Pará (UFPA), Brazil. He earned his Bachelor's degree in Communications Engineering from the same institution, and spent a semester studying abroad at Kagawa University, Japan. He also holds a Master's degree in Machine Learning from UFPA. Mr. Teixeira's current research interests include the application of optimization techniques, computer vision, and machine and deep learning in the context of smart cities.



RAFAEL FOGAROLLI VIEIRA Graduated in computer engineering, specialist in artificial intelligence applied to industry and master in electrical engineering with emphasis in applied computing from the Federal University of Pará (UFPA). Member of the Operational Research Laboratory (LPO) - UFPA. His research interests include evolutionary computing, machine learning, telecommunications networks, and computer networks. He is an Associate Member of the Brazilian Computing Society (SBC).

DANIEL DA SILVA SOUZA obtained a Bachelor's degree in Information Systems in 2016 and a Master's degree in Electrical Engineering with an emphasis on Applied Computing, from the Federal University of Pará (UFPA)in 2018, where he is currently pursuing a Doctoral degree. He is currently a member of the Operational Research Laboratory (LPO). His research areas are focused on Human-Computer Interaction, User Experience, Software Engineering, and Computer Networks. He is an Associate Member of the Brazilian Computer Society (SBC).

IGOR WENNER SILVA FALCÃO has a Bachelor's degreein Information Systems from the Federal Universityof Para (UFPA-2018). Master's degree in ElectricalEngineering from the Graduate Program in ElectricalEngineering (PPGEE), Federal University of Pará(UFPA - 2020). Worked as a scientific initiationscholar by the Institutional Program of ScientificInitiation Scholarships (PIBIC) and as a monitorat the Federal University of Para. Has experiencein Computer Science, focusing on CommunicationNetworks. His research interests are in Communica-tion Networks, Computer Networks, Cloud Computing, and Human-ComputerInteraction. Works at the Operational Research Laboratory (LPO).

DIEGO LISBOA CARDOSO obtained a Bachelor's degree in Computer Science from the University of Amazônia (2002), a Master's degree (2005) and a PhD degree (2010) in Electrical Engineering from the Federal University of Pará and a post-doctoral fellowship at the Royal Institute of Technology of Sweden (KTH). He was Director of Technology at the State Department of Education of the Government of the State of Pará (2008-2009). He was the Rector of Undergraduate Education at the Federal University of South and South-East Pará (UNIFESSPA) (2014). He works as an Associate Professor at the Federal University of Pará in the School of Computer Engineering and Telecommunications and the Graduate Program in Electrical Engineering (PPGEE). He has experience in Computer Science and Computer Engineering, with an emphasis on Performance Evaluation. His work is mainly concerned with the following subject-areas: Digital TV, Access Technologies, Markovian performance and simulation models, applied computational intelligence and optimization techniques.

• • •