electronics

Article

Enhanced Metaheuristic Algorithm-Based Load Balancing in a 5G Cloud Radio Access Network

Krishnamoorthy Suresh [1], Ali Alqahtani [2], Thangaraj Rajasekaran [3], Murugan Suresh Kumar [3], Venugopal Ranjith [1], Raju Kannadasan [4], Nayef Alqahtani [5] and Arfat Ahmad Khan [6,*]

1 Department of Information Technology, Sri Venkateswara College of Engineering, Sriperumbudur 602117, India
2 Department of Networks and Communications Engineering, College of Computer Science and Information Systems, Najran University, Najran 61441, Saudi Arabia
3 Department of Computer Science and Engineering, Sri Venkateswara College of Engineering, Sriperumbudur 602117, India
4 Department of Electrical and Electronics Engineering, Sri Venkateswara College of Engineering, Sriperumbudur 602117, India
5 Department of Agricultural Systems Engineering, College of Agricultural and Food Sciences, King Faisal University, Al-Hofuf 31982, Saudi Arabia
6 Department of Computer Science, College of Computing, Khon Kaen University, Khon Kaen 40002, Thailand
* Correspondence: arfatkhan@kku.ac.th

Abstract: Mobile operators must increase investments in network infrastructures due to the emergent growth of the internet and technological advancements. Mobile operators consider cloud-RAN and software defined networking to be developing technologies that can reduce costs and increase scalability for fifth-generation mobile communication networks (5G). A base station consists of two important components, namely baseband (BBU) and remote radio head (RRH) units. Unbalanced data traffic can arise, leading to call dropping and call blocking. When network traffic conditions start to vary, the performance of the system becomes suboptimal. Self-optimization of the network is necessary to reduce the load of overloaded eNode's with more call blocking, that increase the load of underloaded eNode's with less utilization of resources. The main objective of a self-organizing network is to reduce call blocking and optimize an unbalanced network. The proposed algorithm is an enhanced version of the cat swarm optimization algorithm performed by the host manager entity to select the best BBU-RRH combination after analyzing the quality-of-service (QoS) information from the remaining BBU-RRH configurations. Optimization is carried out on each user after a QoS analysis for every new BBU-RRH combination. The proposed algorithm is implemented in Matlab R2020a and evaluation is conducted in terms of blocking probability, response time, and throughput. The simulation results show that the proposed ECSO optimization algorithm reduces blocking probability by 10%, throughput is increased by 8%, and response time is reduced by 7% as compared with the existing PSO and CSO algorithms.

Keywords: cloud-RAN; enhanced cat swarm optimization algorithm; self-organizing network

check for updates

Citation: Suresh, K.; Alqahtani, A.; Rajasekaran, T.; Kumar, M.S.; Ranjith, V.; Kannadasan, R.; Alqahtani, N.; Khan, A.A. Enhanced Metaheuristic Algorithm-Based Load Balancing in a 5G Cloud Radio Access Network. *Electronics* **2022**, *11*, 3611. https://doi.org/10.3390/electronics11213611

Academic Editor: Christos J. Bouras

Received: 27 September 2022
Accepted: 3 November 2022
Published: 5 November 2022

1. Introduction

1.1. Background

Mobile network operators are encountering a steep rise in data traffic due to the ever-increasing number of mobile users and the demand for high information rates in multimedia applications. Wireless access networks have developed from 3G to 5G networkds to meet demands from mobile gadgets with ubiquitous service access. Small cell organization and high-RF band investigation are two key headings for cutting-edge wireless access networks. With different air interface principles and the multiplication of brilliant gadgets, mobile web traffic is expanding. Likewise, operators are compelled to build capital and operating expenses to address the issues of users. Even then, the average revenue generated from every user does not meet increasing infrastructure costs. Increased traffic demands seem to be challenging and need to be addressed by providing vastly scalable connectivity for increasing IOT-based cellular applications [1,2]. Performance improvement can be achieved by using emerging technologies such as multiple-input multiple-output technology and cloud-RAN with high scalability, flexibility, and millimeter wave [3]. The alternative is to divide larger cells into smaller ones and to exploit the recurrence reiteration, which presents more impedance and expands the expense of framework operation and the executive. The fifth generation (5G) wireless framework will bolster a developing number of mobile gadgets with omnipresent assistance access. To understand 5G networks, ultra-dense small cell organization and distributed computing have been perceived as two key advances. With small cells, the received signal quality is improved on the user side as it is diminished to propel BPS.

The 5G framework implements two emerging technologies, namely: (i) ultra-dense cells which reduce the user's distance with a connected base station enhancing received signal strength and (ii) cloud computing for eliminating inter-cell interference due to ultra-dense small cells. Cloud-RAN virtualizes all base station operations through cloud computing and dynamic load balancing which enhance both spectral and energy efficiency [4,5]. This deals with major issues that include load balancing, security, resource management, and performance monitoring. Load balancing is one of the important issues to be considered which affects computing aid utility and fairness. The QoS has been improved as a result of load balancing by optimizing resource allocation and response time [6]. Users generate various requests with dynamic resource requirements for the execution of their tasks. Hence, cloud data centers require a dynamic outlook to encounter unbalanced loads and to avoid performance deterioration in the QoS. Virtual machine scheduling is implemented in load balancing algorithms by assigning virtual machines to hosts and, in metaheuristic algorithms, the centralized controller controls various processes such as mutation, crossover, and exchanging operations for better assignment of virtual machines to hosts with fewer loads.

Cloud-RAN signifies base station operations virtualized through cloud computing. In cloud-RAN, the baseband and high-layer capacities are executed in concentrated, universally useful processors, as opposed to the neighborhood equipment of the wireless access hubs [7,8]. The access point only holds radio capacities and does not require execution of a conventional full-scale base station layer. This outcome is an another cell engineering approach in which minimal effort hubs are facilitated with wireless access called remote radio heads (RRHs) which are overseen in midway as a concentrated "cloud" or control unit (CU). At the most elevated level, the cloud-RAN idea is observed for network performance virtualization (NPV) strategies [9]. By using high-limit backhaul associations among BS and CB, the cloud-RAN design empowers the encoding and decoding of messages from different cells [10,11]. Since 5G wireless cellular networks are relied upon, bit by bit, to be used with smaller cell sizes to help higher information rate requests, the cloud-RAN design is viewed as a pathway to productive execution when inter-cell obstruction turns out to be progressively significant body-layer interruption [12]. A coordinated multi-point (CoMP) network is otherwise called a multi-input multi-output network (MIMO) framework. Altogether, it can improve the general performance of the cell network [13]. Load balancing in cloud-RAN enables operators to reduce capital costs and operating costs for the multicast network [14,15]. A cloud-RAN 5G framework implements statistical multiplexing and load balancing gains as two important advancements for improvement [16,17]. In this research work, an enhanced cat swarm optimization load balancing algorithm is proposed for the reduction of call blocking [18].

1.2. Literature Review

Several researchers have analyzed different load-balancing techniques for minimizing call blocking. Some of the research articles are reviewed here in detail.

Authors have derived an architectural solution based on computational intelligence and software-defined networks for 5G cloud-RAN. BBU-RRH mapping adjustment has been carried out through load balancing on the prediction of objective and subjective QoE metrics [19]. This mapping was required for the application of UHD video streaming. This intelligent SDN framework predicted the mean opinion score of users transmitting UHD videos within cloud-RAN architecture. Load balancing rules were executed in the SDN framework to maximize QoE in the streaming video download and the BBU pool was generated based on SDN components. The simulation results showed gain attainments of 130% and 60% for conditions of new BBU activation and without activation, respectively. Mahapatra, B. et al. [20] proposed an inter-BBU load-balancing framework that depended on VB's live migration. Two BBUs were considered for migration in a single BBU pool. Resource allocations were assigned on a timely basis with an expected time of completion and VB's migration based on utilization factor. Imbalance load conditions were overcome by increased BBU utilization with proper resource allocation and a load balancing strategy. Server downtime and migration time were reduced by live migration and cloud computing technology.

Some authors proposed a heterogeneous radio access network (H-RAN) with newly developed edge nodes and local processing units present in remote radio heads [21]. An increase in network capacity and a reduction in network latency were impacts of the enhanced remote radio heads. The controlling and computing task of the centralized BBU was accomplished by a heterogeneous radio access network using several VB units in the proposed local processing unit. Load aware dynamic base station switching algorithm was presented for load balancing and power optimization. Performance was compared for both cloud-RAN and H-RAN for equal input packet arrival. The local processing unit and middle edge layer introduction at the remote radio head reduced blocking probability and waiting time as compared with the cloud-RAN architecture. Patel, A. et al. [22] considered a swarm intelligence algorithm for call blocking reduction and load balancing of baseband units. The particle swarm optimization algorithm was implemented with a swarm of initial particles. The candidate solution was determined from each initial particle with random speed and position. Communication between swarms enabled one to find the local best position and swarm at its best, called the global best, among all particles at different positions and speeds. The simulation results proved that blocked calls minimization and equal distribution of load in BBU sectors. QoS is improved in the proposed algorithm by balancing equal load in different sectors of BBU-RRH.

In addition, edge computing on edge infrastructure based on cloud computing technology was proposed by [23]. In order to reduce delay, mobile edge computing offloaded edge devices' computing tasks to edge services available in the network. The challenging task was to design an offload strategy in multi-mobile edge computing and resource-constrained multi-user. In an industrial environment, implementation of task offloading from the resource-constrained edge devices was proposed using particle swarm optimization to edge servers with higher energy efficiency and less delay. Energy consumption, task execution cost, and less time delay were three issues considered for a multi-objective optimization algorithm. Each particle fitness function represented the total cost for offloading tasks to different mobile edge computing servers. The simulation results showed that the proposed particle swarm optimization offloading strategy, as compared with the genetic algorithm and simulated annealing algorithm, reduced the MEC server delay, increased energy consumption balance, and achieved efficient resource allocation.

Further, authors have suggested performance enhancement hierarchical cloud computing architecture using the mobile dynamic cloud [24]. Device-to-device communication was carried out for data transmission in a mobile dynamic cloud. Mobile network overall capacity was increased through channel utilization and traffic overloading over device-to-device communication links. The amount of offloaded data depended on the number of devices in the cloud and content matching values which increased the capacity of the mobile network by 10%. Pompili et al. [25] proposed cloud-RAN with all base station resources accumulated as a centralized pool, and frequent communication, cooperation, and spectrum resources were dynamically allocated based on dynamic needs. The proposed work explained, in detail, how to overcome the limitations of current wireless networks such as the implementation of spatial multiplexing and macro diversity. The allocation of virtual base stations and mitigation of inter-cell inference problems were discussed in detail. Liu et al. [26] discussed the effect of front haul capacity on cloud-RAN performance. Throughput was maximized through the implementation of efficient compression and quantization techniques in the front haul. Uniform scalar quantization was suggested for reducing the complexity of uplink communication of the OFDMA-based cloud-RAN system. Orthogonal subcarriers were assigned for every mobile user before transmission and applied with joint wireless power control and front haul quantization. A joint optimization algorithm improved gain performance over wireless power control and quantization. The proposed uniform quantization achieved better throughput efficiency in cloud-RAN. Liu et al. [27] proposed a clustering scheme for operation in a semi-dynamic manner over large-scale channel information with low complexity. Clustering was changed to coordinated beam forming and joint transmission mode when the backhaul capacity was unlimited and stringent. This clustering scheme was applied for a downlink cloud radio access network in which all base stations were connected to a central unit through backhaul links with limited capacity. A hybrid coordinated multipoint transmission mode was considered for maximizing throughput with the constraint on backhaul capacity and training overhead. Boulos et al. [28] implemented a cloud-RAN architecture with base stations divided into BBU and RRH units. In this architecture, BBU units were accommodated in a single pool and RRHs were distributed across multiple points. One-to-many logical mapping was established between BBU and RRH for reducing network power consumption. The RRH clustering problem was avoided by assigning many RRHs to one BBU and optimal heuristic solutions were found without QoS degradation.

Moreover, design of cloud-RAN based on the quality of experience model has been suggested. Enhanced cloud-RAN development was based on passive optical networks which exploited power over fiber [29]. This installation was known for its low cost with the operation of RRH without any external power supply. This new proposed design with several RRHs, optical line terminals, and sleep schedule ensured an enhanced network to satisfy the users expected quality of experience. Liu et al. [30] proposed a network coding technique in a cloud-RAN model for multicasting messages to neighboring RRHs and beamforming vectors. Successive convex approximation and sparse optimization were applied over the front haul network in network coding design. A data-sharing strategy was adopted in which users' messages were broadcast from a central processor to all RRHs connected to it. Optimization of the routing strategy over front haul network approximation was performed based on successive convex approximation. Multicast was more efficient in multi-hop front haul than unicast of cloud-RAN. Wang et al. [31] dealt with power consumption issues on a mobile service provider in which cloud-RAN and mobile edge cloud computing were integrated. In this paper, a unifying framework was designed for mobile service provider power performance tradeoff carried out by proper allocation of computation resources in mobile edge cloud computing and network resources in cloud-RAN. The extended Lyapunov technique was used for developing a new optimization framework and resource scheduling was considered stochastic. A new algorithm called VariedLen was designed for consecutive job requests with variable lengths and the application of dynamic conditions in mobile service. The time average profit achieved in the proposed algorithm was nearer to the optimum diminishing gap for MSP with low congestion and more stability attained.

Further, in another study, the authors implemented a dynamic greedy spike load balancer design for a cloud-RAN network [32]. This load balancer design was enabled with user-level virtualization. Hence, new users could be allocated to different hosts in cloud, independent of their origin base station. New users were allocated to different hosts in virtualized user level RAN architecture for reducing interference and isolation improvement using open air interface for baseband processing. In addition, Ari et al. [33] focussed on minimizing network costs and balancing network load with a proposed resource allocation method. The mapping between the user/remote radio head and the remote radio head/baseband unit was an NP-hard problem. Optimization to decompose the two mappings was carried out using the artificial bee colony and ant colony algorithms. The combination of the two algorithms resulted in the proposed Bee-Ant cloud-RAN with improved spectral efficiency and throughput. Finally, the authors implemented a framework for cloud-RAN called deep reinforcement learning [34]. The authorized DRL agent activated three states: defined state, action, and reward function on the remote radio head. Beam forming was transmitted at active RRHs in a regular period of time for achieving high sum rate in a time varying network.

Based on the robust literature review, the objectives of this article are described as follows:

1.3. Objectives

- To achieve minimum call blocking in the network and to solve the load balancing optimization problem with the proposed enhanced cat swarm optimization algorithm;
- To identify suitable BBU-RRH configuration by the host manager based on collected information from all the RRHs in the network;
- To analyze the QoS information of the current BBU-RRH configuration and to utilize its optimization step to analyze the QoS of each candidate solution for the new BBU-RRH configuration;
- To evaluate the performance of this proposed approach in terms of blocking probability, response time, and throughput.

2. Cloud-RAN System Architecture

The main objective of the method proposed in this research work is to solve the load balancing problem and to reduce call blocking in the network. An enhanced cat swarm algorithm is proposed for load-balancing optimization. The system model of cloud-RAN is shown in Figure 1.

In Cloud-RAN, cloud computing improves the 5G network framework operation. This system model consists of virtual BBUs, BBUs, cloud controllers, RRHs, and general purpose processors (GPPs). Digital IQ signals are conveyed through a common public radio interface to RRHs. BBUs in the pool are connected with each other and controlled by a host manager. The load on each BBU is regulated by the host manager who is responsible for the selection of the proper BBU-RRH configuration. Multiple sectors are handled by each BBU at a time and multiple RRHs are present in each sector with BBU resources entirely utilized. Each RRH is available in one sector during any period and the BBU load is calculated based on the total number of users actively present. The hard capacity of a sector is defined as the limitation on the number of active users based on hardware or software limitations [26,27].

BBU pool, otherwise referred to as a data processing center, consists of shared, centralized, and virtualized functioning units. Based on dynamic user demand, resources are allocated to virtual BBUs interconnected with each other. Multiple RRHs are supported by the BBU pool. BBUs are interconnected with each other through the X2 interface. A BBU is connected to the mobile network through the S1 interface called the backhaul link. Real-time virtualization is applied to high-performance processors present in BBUs. BBUs are installed on virtual machines in the cloud data center over physical computing cores.

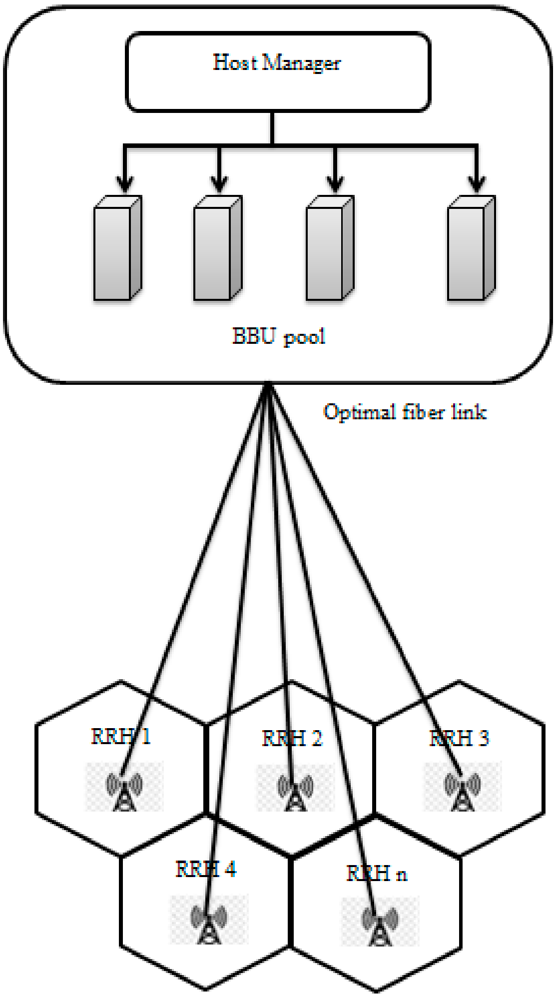

Figure 1. Cloud-RAN system model.

The distribution of the RRHs is located in the service area where mobile users' communication services are spread. The area that extends up to the connection between the user and RRH maintaining the required QoS is called the coverage area. The two parts of an RRH are the transmitter and receiver. Digital signals are received in the transmitter through the CPRI interface and converted to analog. RF frequency up conversion is the next step in the process. Amplification and filtration are performed, and the output is sent via the antenna. The receiver receives the antenna signal, filters, and amplifies, and if frequency down conversion occurs, digital conversion is performed before transmitting to fiber link processing through CPRI.

Regarding fiber links, each set of RRHs is connected to a BBU employing high bandwidth and low latency links for handling multiple RRH requirements. Millimeter wave communication, optical fiber communication, and cellular communication are different technologies involved in the link setup process. The bandwidth available in highly expansive optical fiber communication is large, whereas millimeter wave communication or cellular communication is less expensive in deployment with more latency.

Regarding dynamic traffic load adaption, mobile users' real-time movement produces a dynamic change in traffic loads throughout the day. This results in certain base stations becoming overloaded while the others are in an idle state and, in turn, this results in improper utilization of radio resources and wastage of processing power in large quantities. Low-traffic load RRHs are grouped and controlled by one BBU and high-traffic load RRHs are allocated to multiple BBUs based on user demands. Statistical multiplexing gain is increased significantly as the overall utilization rate is effectively utilized [28].

Load balancing is performed both on the BBU and RRH. The BBUs are organized as a single BBU unit pool and load balancing is achieved through the allocation of BBU resources within the pool. Load balancing on the RRH is attained depending upon BBU's capacity within the pool which changes as the user moves to different cells. Changes in the BBU (or) RRH occur based on network traffic load. The BBU currently associated with the RRH is turned off when the RRH traffic is low and connected to an alternative BBU within the pool. Based on user demand, the RRH is switched on or off and connected with another RRH within the pool.

Load Balancing in the 5G Cloud Radio Access Network

The 5G network considers load balancing for real-time carrier-grade services in cloud-RAN. An enhanced cat swarm optimization algorithm is proposed for consideration. eNodeB control center controls all layer-2 (L2) control applications and stores data of a considerable number of users. Multiple applications run on the L2 layer controlled by a single control center, and based on the user's demands, frequency bands are allocated. A user equipment context is the format for demands from users.

Pooled virtual machines hosted by the server are used for processing instead of a single machine for each cell in the BBU pool of cloud-RAN. Each virtual machine is assumed to be allocated to its local cache. User equipment context is stored in the local cache for every virtual machine for fast and reduced latency access. Load balancing is achieved only if the user equipment context is allotted to the correct virtual machine to enable a reduction in the accessing and processing time of the user equipment context. Many complex and intelligent load-balancing algorithms have been proposed for various networked frameworks and distributed computing. The objective of this research work is to comprehend the attributes of real network traffic load as far as the user equipment context, in order to prepare for cloud-RAN in the 5G network [28,29]. An enhanced cat swarm optimization algorithm (ECSO) has been used for solving the problems seen in load balancing (Figure 2).

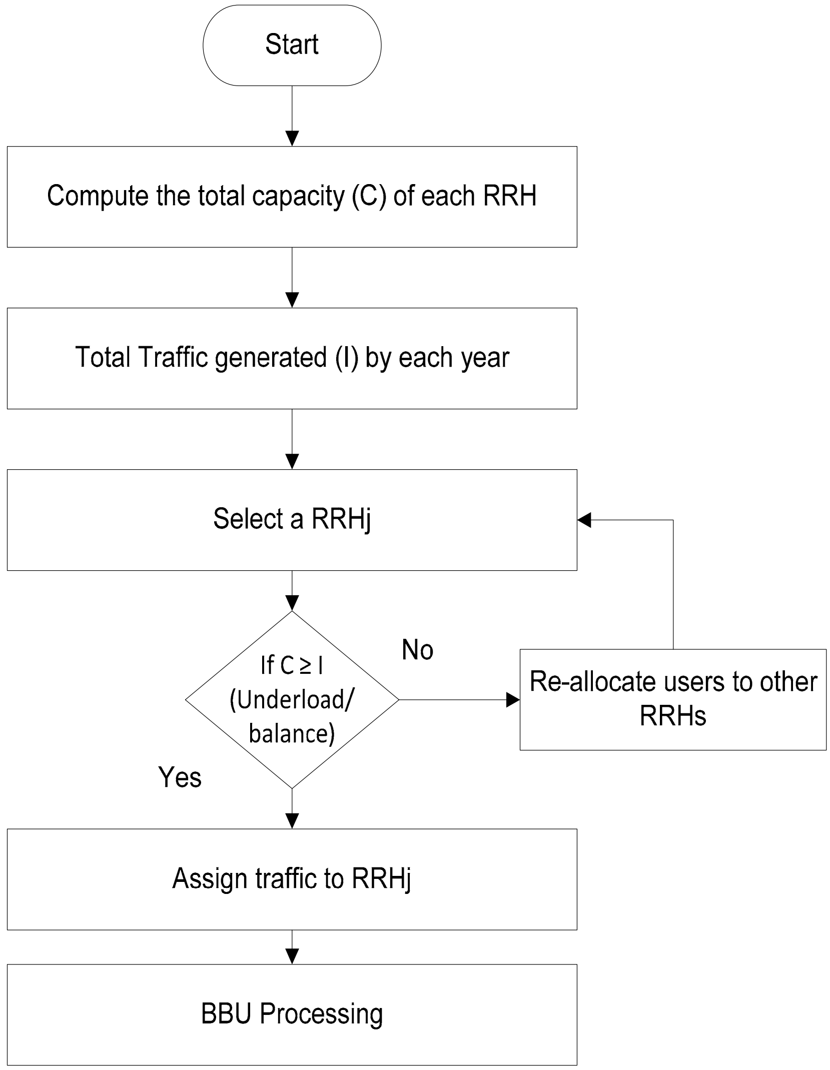

Figure 2. Load balancing in CRAN.

3. Enhanced CAT Optimization Algorithm Based Load Balancing in the 5G Cloud Radio Access Network

In the proposed algorithm, two major behaviors of cats are modeled considering two submodels likely tracking mode and seeking mode. By adapting the mingling process of these two modes with a user-defined proportion, ECSO can present better performance. The number of cats is the first step to be decided in ECSO, and these cats are employed in the proposed algorithm for problem-solving mechanisms. Specifically, M dimensions are considered for each cat position and its every dimension has its velocity and fitness value to represent cat accommodation for the flag identification and fitness function for cat check-in seeking or tracing mode [25,26]. Further, increment iterations can be performed untill the best position in one of the cats is determined [27,28].

The ECSO algorithm is a single objective and continuous algorithm that is influenced by resting and tracing cat behaviors. Despite spending maximum time at rest, the cats are very conscious of happenings in their surroundings. When the target is seen, they quickly start moving toward it. Two modes of the ECSO algorithm, namely tracing and seeking modes, are mathematically modeled step-by-step, as described below [29,30]:

Initialization Fitness value, flag, and position are identified and the solutions for each cat's M dimensions for each position are provided in the whole search space with its velocity and fitness value. These values are clarified up to the extent of the solution set which is perfect and the cats are classified in any one of two modes, i.e., seeking or tracing modes.

Fitness calculation Once a suitable position is determined, fitness calculation is done for different positions as given:

$$\text{Fitness Calculation} = \text{Max}(\text{QoS}) \tag{1}$$

then, network service quality is calculated based on a key performance indicator such as reverted calls in the network. The hard capacity (HC) of each sector is increased when an active number of users causes a rising network to block calls. The key performance indicator for blocked calls is given by:

$$KPI_{BC} = \left[\sum_{s=1}^{i}\left(\left(\sum_{j=1}^{M} AC_j S_{js}^{t+1}\right) - H^c\right)\right]^{-1} \tag{2}$$

where the binary variable S_{js}^{t+1} is equal to 1 if j^{th} remote radio head belongs to sectors. AC_j is represented by the active users served by j^{th} remote radio head while H^c is represented as the hard capacity of sectors. Furthermore, the maximum QoS function is assumed to be equal to the key performance indicator metric of the blocked calls [31].

The objective to achieve maximum QoS functionality is given by:

$$QoS_{max} = KPI_{BC} = \left[\sum_{s=1}^{i}\left(\left(\sum_{j=1}^{M} AC_j S_{js}^{t+1}\right) - H^c\right)\right]^{-1} \tag{3}$$

Each RRH is assigned to an individual sector at time instant $t + 1$ for the binary variable $\sum_{s} S_{js}^{t+1} = 1$. The new BBU-RRH configuration in search space 2NS is determined if the binary variable S_{js}^{t+1} is varying where the number of RRHs is N and the number of sectors is S. The search space is reduced to SN by defining a new variable, V_j^{t+1} is defined such that $V_j^{t+1} = \left\{V_1^{t+1},\ V_2^{t+1},\ V_3^{t+1} \ldots . V_N^{t+1}\right\}$. Where V_j^{t+1} = J denotes that the j^{th} remote radio head is connected to the sector.

Seeking mode After the search mode, the cat finds out its position with some changes. The sequence of steps in the seeking mode is:

- The individual position of each cat is represented.
- Visual memory pool copy locations are remembered.

The mutation function formula is used with a few slight changes in the seeking memory pool (SMP). The aeration mutation-based purification formula is given by:

$$Y[k] = Y[k] + \nabla Y[k] \tag{4}$$

The situation is updated for each cat on choosing the SMP point based on best fitness.

Tracing mode In the revelation mode, cats are redirected to particles in the PSO algorithm and they are regenerated to its own speed, positioned by the cat's selection of its best position. The velocity and position of the j^{th} cat in the dimension D are given by:

$$Position_k = (Position_{k1,} Position_{k2,} \ldots , Position_{kD}) \tag{5}$$

$$Velocity_k = (Velocity_{k1,} Velocity_{k2,} \ldots , Velocity_{kD}) \tag{6}$$

where P represents the position of the cat and V represents the cat's velocity. The global best position of the cat swam represented is given below:

$$gbest = (gbest_1, gbest_2, \ldots gbest_D) \tag{7}$$

The updated equations are given below:

$$V_{kD} = \delta \times V_{kD} + A \times I(gbest_D - P_{jD}) \tag{8}$$

$$P_{kD} = P_{kD} + V_{kD} \tag{9}$$

where A is represented as the acceleration constant, I is denoted as the random number in the selection range from 0 to 1, j^{th} iteration, and G best is represented as the global condition of a cat with excellent fitness.

Termination criteria: The termination criteria include the iteration's maximum value and the process is stopped after determining the best position for the cat [32,33].

The enhanced cat swarm optimization implementation algorithm is explained as given below and the same has been illustrated in Figure 3:

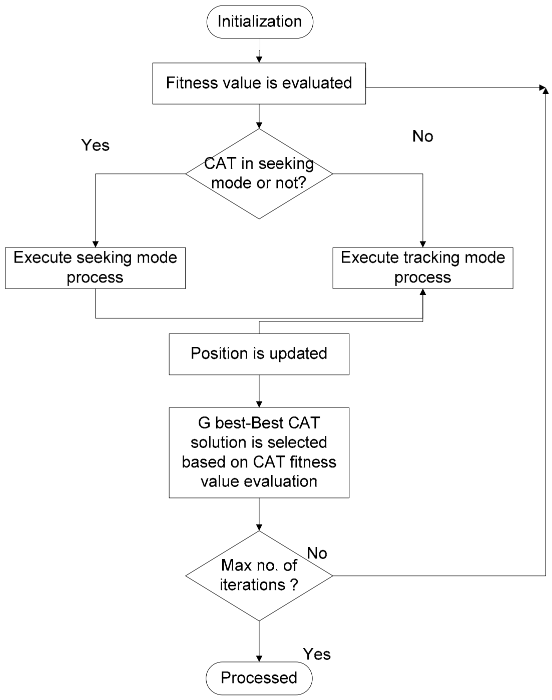

Figure 3. Flowchart of the enhanced cat swarm optimization algorithm.

Step 1: Create N cats in the process.

Step 2: Randomly sprinkle the cats into the D = (N/2 + 1) number of h(n) coefficients dimensional solution space within (2, +2) and randomly give values, which are within (0.1, 0.1), to the velocity of every cat. Then, haphazardly pick the number of cats and set them into tracing mode according to MR, and the others are set into seeking mode.

Step 3: Evaluate the fitness value of each cat by applying the positions of cats to the fitness function, which represents the criteria of our goal, and keeps the best cat in memory. Note that it is only needed to remember the position of the best cat (gbest) that represents the best solution so far.

Step 4: Move the cats according to their flags; if the cat is in seeking mode, apply the cat to the seeking mode process, otherwise, apply it to the tracing mode process.

Step 5: Re-pick the number of cats and set them into tracing mode according to MR, then set the other cats into seeking mode.

Check the termination condition, if satisfied, terminate the program and otherwise repeat Steps 3–5.

4. Results and Discussions

This section deals with the proposed method simulation results obtained and analyzed using MATLAB R2020a. The simulation parameter of the proposed method is represented in Table 1. Three sectors are handled by each BBU and the number of active users in each cell ranges from a minimum of 40 to a maximum of 70. In addition, the hard capacity of each sector is considered to be 150 and a smooth traffic shaper is implemented that blocks traffic flow when it exceeds the allotted limit. The proposed method is examined for one hexagon cell with the presence of 10 RRHs. The idle power of GPP is considered to be 120 Watts, the GPP maximum power is 215 Watts, and 75 users are available within the cell [35,36].

Table 1. Simulation parameter.

Simulation Parameter	Values Assumed
Cells considered	1
Cell shape	Hexagonal
RRHs Number	10
GPPIdle power	120 W
GPPMaximum power	215 W
Bandwidth	20 MHz
Users Count	75

Different performance metrics are considered in this work, notably blocking probability, response time, and throughput are simulated and illustrated in Figures 4–6, respectively to assess the performances.

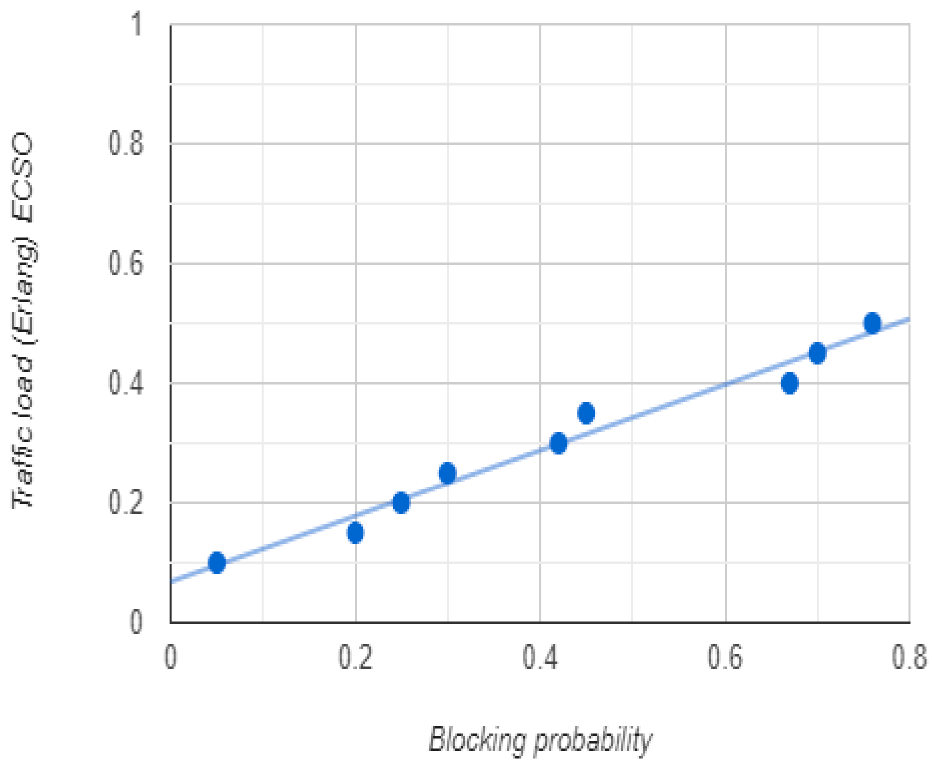

Figure 4. Blocking probability versus traffic load.

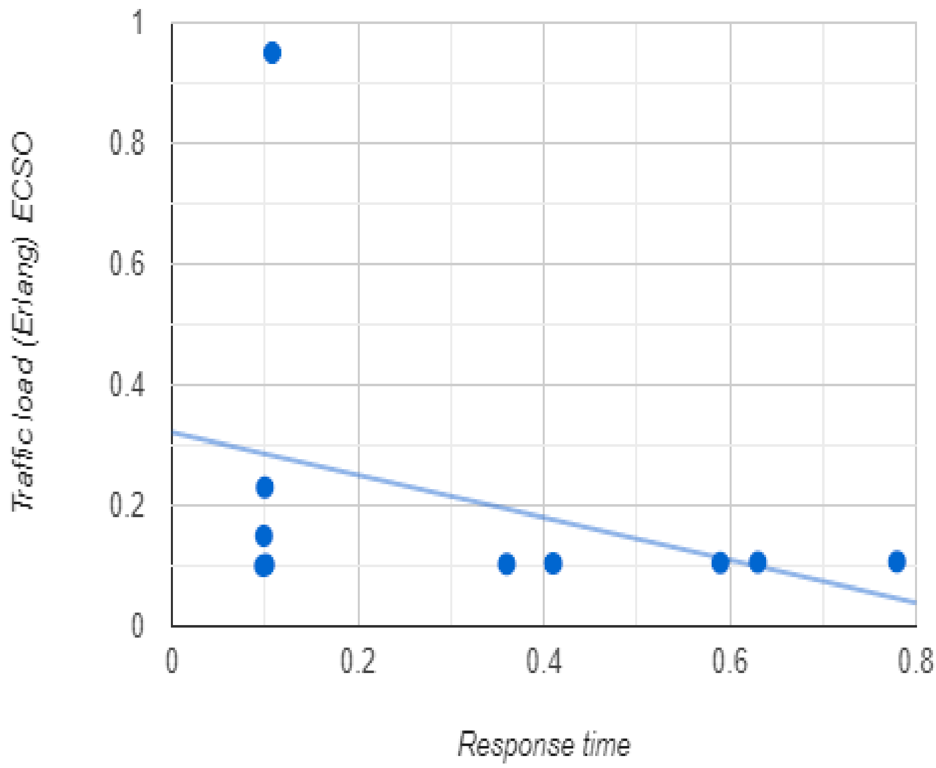

Figure 5. Traffic load against response time.

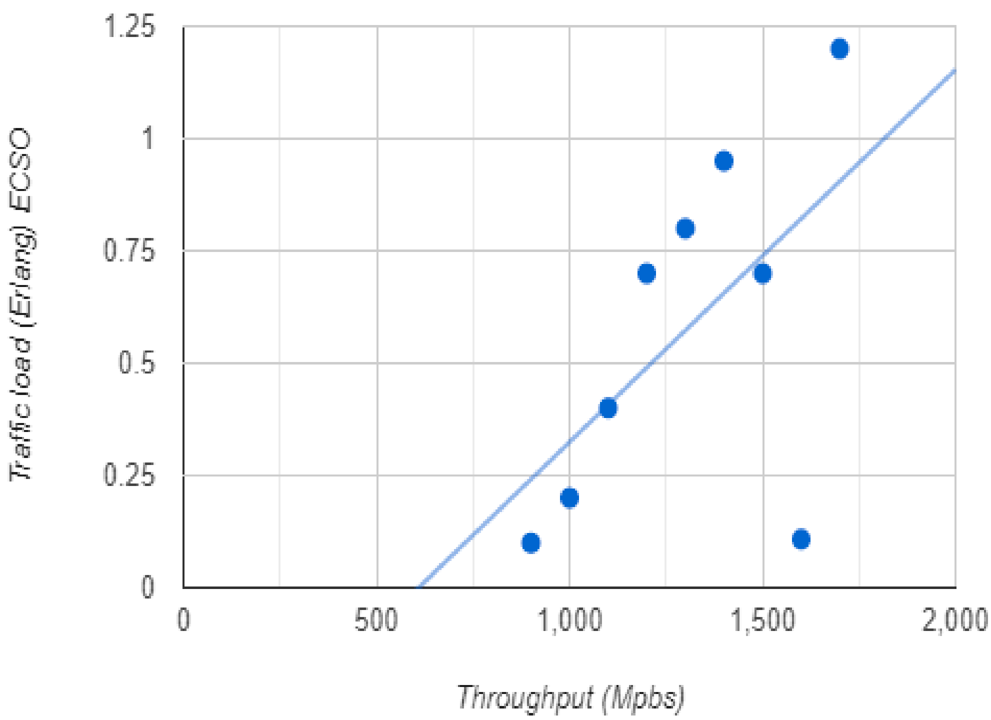

Figure 6. Traffic load against throughput.

Blocking probability is a measure of the number of blocked service requests (Sbs) to received service requests (Srs):

$$\text{BlockingProbability} = \frac{S_{bs}}{S_{rs}} \tag{10}$$

Response time measured in milliseconds (ms) is calculated based on the time difference between the request for service from the user and the response time for the incoming request. The transmission of data calculated in a particular time duration is called throughput measured in Mbps and is given by:

$$\text{Throughput} = \frac{\text{Transmitted data byte}}{\text{Time taken for transmission}} \tag{11}$$

The proposed ECSO optimization algorithm to find the global best cat position with varying cat velocity is evaluated for blocking probability ranging from 0.05 to 0.5. Traffic load varies from 0.1 to 0.9 with reduced blocking probability as shown in Table 2. The ECSO optimization algorithm shows better response time calculated for traffic load intensity ranging from 0.1 to 0.95, as shown in Table 3. Throughput is increased from 900 to 1700 Mbps, calculated for traffic load intensity ranging from 0.1 to 1.2, as shown in Table 4.

Table 2. Blocking probability versus traffic load.

Blocking Probability	Traffic Load		
	PSO	CSO	ECSO
0.05	0.22	0.15	0.1
0.1	0.53	0.39	0.2
0.15	0.57	0.46	0.25
0.2	0.61	0.47	0.3
0.25	0.66	0.53	0.42
0.3	0.71	0.55	0.45
0.35	0.78	0.72	0.67
0.4	0.85	0.77	0.70
0.45	0.89	0.80	0.76
0.5	1.1	1.0	0.9

Table 3. Traffic load against response time.

Response Time	Traffic Load		
	PSO	CSO	ECSO
0.098	0.45	0.38	0.1
0.099	0.54	0.41	0.15
0.1	0.66	0.53	0.23
0.101	0.72	0.60	0.29
0.102	0.77	0.68	0.36
0.103	0.83	0.75	0.41
0.104	0.89	0.81	0.59
0.105	0.95	0.90	0.63
0.106	1.03	0.99	0.78
0.107	1.10	1.05	0.81
0.108	1.15	1.08	0.95

Table 4. Traffic load against throughput.

Throughput (Mpbs)	Traffic Load		
	PSO	**CSO**	**ECSO**
900	0.05	0.08	0.1
1000	0.12	0.15	0.2
1100	0.23	0.39	0.4
1200	0.59	0.65	0.7
1300	0.64	0.73	0.8
1400	0.78	0.81	0.95
1500	0.81	0.94	0.7
1600	0.90	0.99	0.108
1700	1	1.1	1.2

The proposed method uses the ECSO optimization algorithm, whereas in existing methods, the CSO and PSO algorithms have been used. A comparative analysis of the proposed method against the existing method of blocking probability against traffic load is shown in Figure 7.

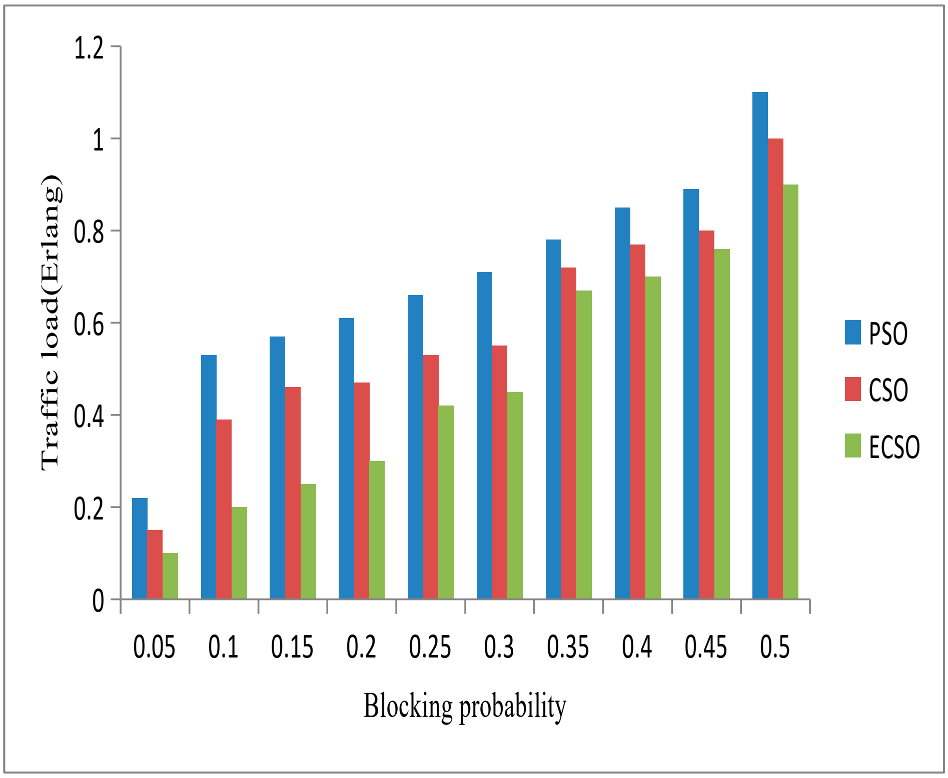

Figure 7. Comparative analysis of blocking probability against existing methods.

Figure 7 represents the blocking probability comparison between the proposed method and existing methods such as CSO and PSO. The blocking probability of the experimental analysis ranges from 0.05 to 0.5. When the blocking probability increases, the traffic load also increases. For a blocking probability of 0.05, the traffic load of the proposed method is 0.1 but the traffic loads of the existing methods are 0.22 (PSO) and 0.154 (CSO). When the traffic load of the proposed method is 0.25, the traffic loads of the existing methods are 0.57 and 0.46, and the blocking probability is 0.15. The blocking probability of the proposed method is 0.2, when the traffic loads of the proposed method and the existing methods are

0.3, 0.47, and 0.61, respectively. When the blocking probability is 0.3, the traffic load varies among the proposed and the existing methods, i.e., 0.71, 0.55, and 0.45 respectively. Clearly, improved results are obtained for the proposed method as compared with the existing approaches. A comparative analysis of the proposed method and the existing methods of response time against the traffic load is shown in Figure 8.

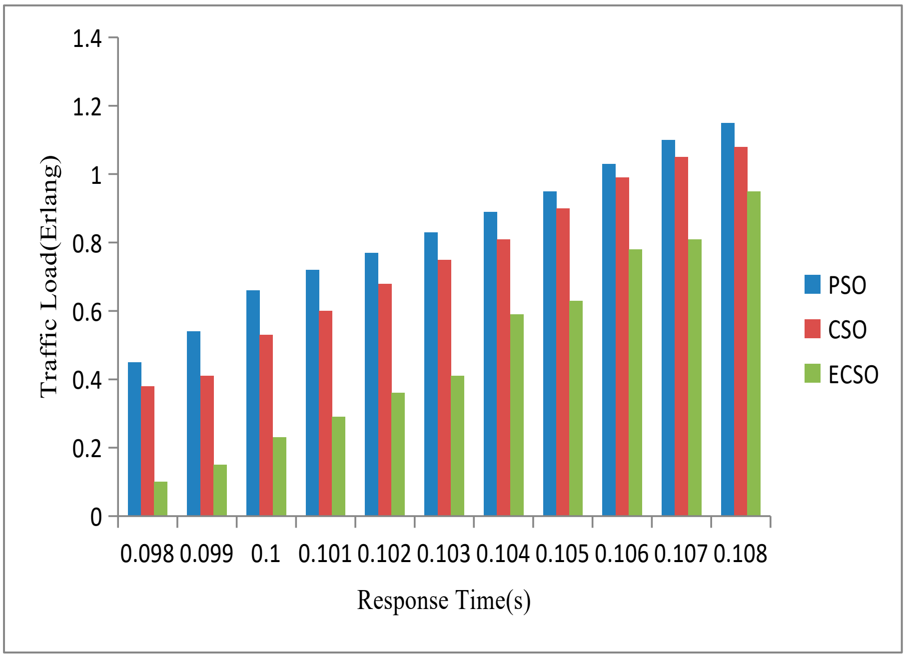

Figure 8. Comparative analysis of response time against existing methods.

Further, a comparative analysis of the proposed and the existing methods throughput against the traffic load function is shown in Figure 9. It represents the comparative analysis of the proposed and the existing methods against the throughput. Throughput is estimated for the whole network and is calculated at the BBU pool. The throughput of the experimental analysis varies from 900 to 1700 Mpbs. When the comparative analysis of the throughput is 900 Mpbs, the traffic load of the proposed and the existing methods are 0.1, 0.05, and 0.08 respectively. When the traffic load of the proposed method is 0.4, the traffic loads of the existing methods are 0.23 and 0.39, respectively, and the throughput is 1100 Mpbs. When the throughput is 1300 Mbps, the traffic loads of the proposed and the existing methods are 0.8, 0.64, and 0.73, respectively. When the traffic load of the proposed method is 0.7, the traffic loads of the existing methods are 0.81 and 0.94, and the throughput is 1500 Mpbs. When the throughput is 1700 Mpbs, the traffic loads of the proposed and existing methods are 1.2, 1, and 1.1, respectively. As compared with the existing methods, the proposed method of throughput is much better than the existing method.

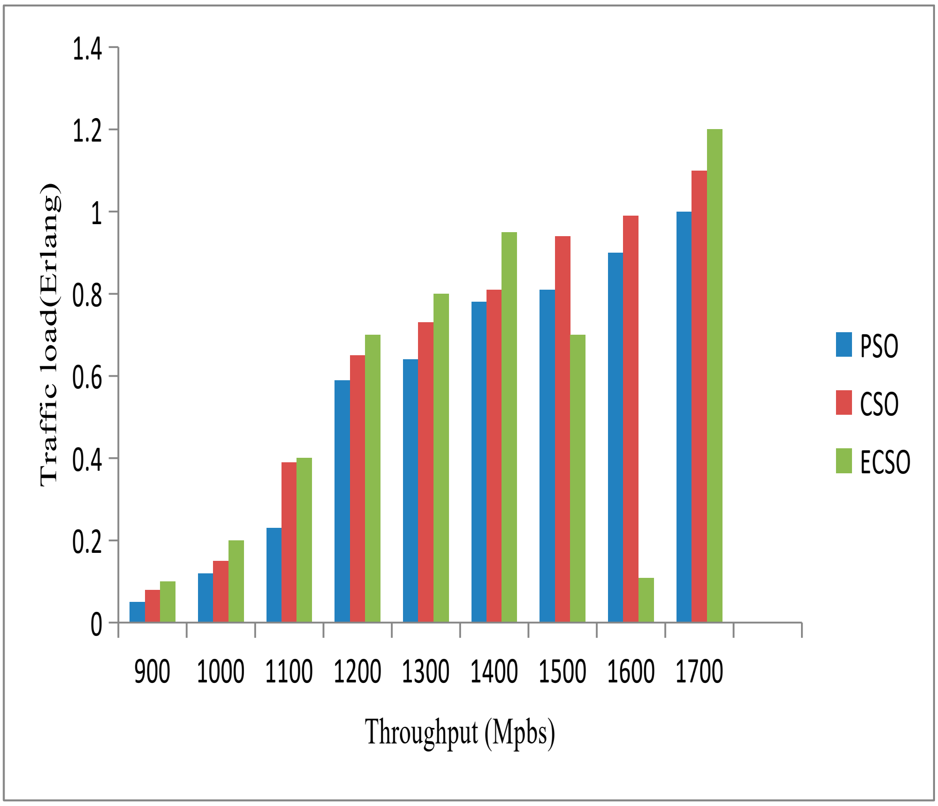

Figure 9. Comparative analysis of throughput against existing methods.

5. Conclusions

The proposed work achieves minimum blocked calls and the load balancing problem is solved with the implementation of an enhanced cat swarm optimization algorithm. QoS maximization is attained in this algorithm with improved call blocking probability. The existing BBU-RRH configuration is analyzed for the QOS calculation and further used in the optimization process for the selection of new BBU-RRH configurations for the individual user. This enhanced CSO was used to achieve minimum call blocking and maximize QoS. The proposed method's experimental results are compared with the PSO and CSO optimization algorithms to prove improved performance in terms of throughput, response time, and blocking probability. The blocking probability is reduced by 10% in the ECSO optimization algorithm as compared with the PSO and CSO algorithms. Throughput is increased by 8% in the ECSO optimization algorithm, ranging from 900 to 1700 Mbps as compared with existing algorithms. The response time is 7% less in the proposed algorithm as compared with the existing algorithms for increasing traffic load. In future work, load balancing optimization for the BBU-RRH combination can be implemented with the latest computing algorithms of hybrid nature and turning on/off BBU units for grid energy consumption optimization.

Author Contributions: Conceptualization, K.S., A.A., and T.R.; methodology, K.S.; software, K.S., M.S.K. and V.R.; validation, R.K., A.A., N.A., and A.A.K.; formal analysis, A.A., K.S., and T.R.; investigation, A.A.K.; resources, T.R.; data curation, M.S.K. and V.R.; writing—original draft preparation, K.S. and R.K.; writing—review and editing, N.A. and A.A.K.; visualization, K.S.; supervision, R.K.; project administration, A.A.K.; funding acquisition, A.A. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by Department of Computer Science, College of Computing, Khon Kaen University, Thailand.

Acknowledgments: We are extremely thankful to the Department of Computer Science, College of Computing, Khon Kaen University, Thailand for supporting this work.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

BBU	Baseband unit
CSO	Cat swarm optimization
Cloud-RAN	Cloud Radio Access Network
COMP	Coordinated multipoint transmission
CMRI	Common Public Radio Interface
ECSO	Enhanced Cat Swarm Optimization Algorithm
GPP	General Purpose Processors
IOT	Internet of Things
KPI	Key Performance Indicators
LAN	Local Area Network
MIMO	Multiple Input Multiple Output
NPV	Network Performance Virtualization
PSO	Particle Swarm Optimization
QoS	Quality of service
RRH	Remote Radio Head

References

1. Wu, J.; Zhang, Z.; Hong, Y.; Wen, Y. Cloud radio access network (C-RAN): A primer. *Inst. Electr. Electron. Eng. Netw.* **2015**, *29*, 35–41. [CrossRef]
2. Dai, B.; Yu, W. Sparse Beamforming and User-Centric Clustering for Downlink Cloud Radio Access Network. *Inst. Electr. Electron. Eng. Access* **2014**, *2*, 1326–1339. [CrossRef]
3. Simeone, O.; Maeder, A.; Peng, M.; Sahin, O.; Yu, W. Cloud radio access network: Virtualizing wireless access for dense heterogeneous systems. *J. Commun. Netw.* **2016**, *18*, 135–149. [CrossRef]
4. Zhou, Y.; Yu, W. Optimized Backhaul Compression for Uplink Cloud Radio Access Network. *IEEE J. Sel. Areas Commun.* **2014**, *32*, 1295–1307. [CrossRef]
5. Tang, J.; Tay, W.P.; Quek, T.Q.S. Cross-Layer Resource Allocation With Elastic Service Scaling in Cloud Radio Access Network. *IEEE Trans. Wirel. Commun.* **2015**, *14*, 5068–5081. [CrossRef]
6. Suresh, K.; Kumaratharan, N. SDN Controller Allocation and Assignment based on Multicriterion Chaotic Salp Swarm Algorithm. *Intell. Autom. Soft Comput.* **2021**, *27*, 89–102. [CrossRef]
7. Hung, S.-C.; Hsu, H.; Lien, S.-Y.; Chen, K.-C. Architecture Harmonization Between Cloud Radio Access Networks and Fog Networks. *IEEE Access* **2015**, *3*, 3019–3034. [CrossRef]
8. Tran, T.X.; Kazemi, F.; Karimi, E.; Pompili, D. Mobee: Mobility-aware energy-efficient coded caching in cloud radio access networks. In Proceedings of the IEEE 14th International Conference on Mobile Ad Hoc and Sensor Systems, Orlando, FL, USA, 22–25 October 2017; pp. 461–465.
9. Suresh, K.; Kumaratharan, N. Performance Modelling of Service Function Chaining in Distributed Controllers Secure Black SDN with NFV Architecture. *Solid State Technol.* **2020**, *63*, 7558–7567.
10. Li, J.; Peng, M.; Cheng, A.; Yu, Y.; Wang, C. Resource allocation optimization for delay-sensitive traffic in fronthaul constrained cloud radio access networks. *IEEE Syst. J.* **2014**, *11*, 2267–2278. [CrossRef]
11. Ugur, Y.; Awan, Z.H.; Sezgin, A. Cloud radio access networks with coded caching. In Proceedings of the 20th International ITG Workshop on Smart Antennas, Munich, Germany, 9–11 March 2016; pp. 1–5.
12. Khan, M.; Alhumaima, R.S.; Al-Raweshidy, H.S. QoS-Aware Dynamic RRH Allocation in a Self-Optimized Cloud Radio Access Network With RRH Proximity Constraint. *IEEE Trans. Netw. Serv. Manag.* **2017**, *14*, 730–744. [CrossRef]
13. Tran, T.X.; Le, D.V.; Yue, G.; Pompili, D. Cooperative Hierarchical Caching and Request Scheduling in a Cloud Radio Access Network. *IEEE Trans. Mob. Comput.* **2018**, *17*, 2729–2743. [CrossRef]
14. Liu, L.; Patil, P.; Yu, W. An uplink-downlink duality for cloud radio access network. In Proceedings of the 2016 IEEE International Symposium on Information Theory (ISIT), Barcelona, Spain, 10–15 July 2016; pp. 1606–1610.
15. Wang, X.; Wang, K.; Wu, S.; Di, S.; Yang, K.; Jin, H. Dynamic resource scheduling in cloud radio access network with mobile cloud computing. In Proceedings of the IEEE/ACM 24th International Symposium on Quality of Service (IWQoS), Beijing, China, 20–21 June 2016; pp. 1–6.
16. Niu, B.; Zhou, Y.; Shah-Mansouri, H.; Wong, V.W. A dynamic resource sharing mechanism for cloud radio access networks. *IEEE Trans. Wirel. Commun.* **2018**, *12*, 8325–8338. [CrossRef]
17. Marotta, M.A.; Kaminski, N.; Gomez-Miguelez, I.; Granville, L.Z.; Rochol, J.; DaSilva, L.; Both, C.B. Resource sharing in heterogeneous cloud radio access networks. *IEEE Wirel. Commun.* **2015**, *22*, 74–82. [CrossRef]

18. Karneyenka, U.; Mohta, K.; Moh, M. Location and mobility aware resource management for 5G cloud radio access networks. In Proceedings of the International Conference on High Performance Computing & Simulation (HPCS), Genoa, Italy, 17–21 July 2017; pp. 168–175.
19. Medeiros, G.O.; Costa, J.C.W.A.; Cardoso, D.L.; Santos, A.D.F. An Intelligent SDN Framework Based on QoE Predictions for Load Balancing in C-RAN. *Wirel. Commun. Mob. Comput.* **2020**, *2020*, 7065202. [CrossRef]
20. Mahapatra, B.; Turuk, A.K.; Panda, S.K.; Patra, S.K. Utilization-aware VB migration strategy for inter-BBU load balancing in 5G cloud radio access networks. *Comput. Netw.* **2021**, *181*, 107507. [CrossRef]
21. Mahapatra, B.; Turuk, A.K.; Ray, N.; Yadav, A.K. H-RAN: An Approach toward Cloud-RAN Load Balancing. In Proceedings of the International Conference on Information Technology (ICIT) 2018, Bhubaneswar, India, 19–21 December 2018; pp. 216–220.
22. Patil, A.; Gala, H.; Kapoor, J. Dynamic Load Balancing in Cloud Computing using Swarm Intelligence Algorithms. *Int. J. Comput. Appl.* **2015**, *15*, 15–21.
23. You, Q.; Tang, B. Efficient task offloading using particle swarm optimization algorithm in edge computing for industrial internet of things. *J. Cloud Comput. Adv. Syst. Appl.* **2021**, *10*, 1–11. [CrossRef]
24. Jo, M.; Maksymyuk, T.; Strykhalyuk, B.; Cho, C.H. Device-to-device-based heterogeneous radio access network architecture for mobile cloud computing. *IEEE Wirel. Commun.* **2015**, *3*, 50–58. [CrossRef]
25. Pompili, D.; Hajisami, A.; Viswanathan, H. Dynamic provisioning and allocation in Cloud Radio Access Networks (C-RANs). *Ad Hoc Netw.* **2015**, *30*, 128–143. [CrossRef]
26. Liu, L.; Bi, S.; Zhang, R. Joint Power Control and Fronthaul Rate Allocation for Throughput Maximization in OFDMA-Based Cloud Radio Access Network. *IEEE Trans. Commun.* **2015**, *63*, 4097–4110. [CrossRef]
27. Liu, D.; Han, S.; Yang, C.; Zhang, Q. Semi-dynamic User-Specific Clustering for Downlink Cloud Radio Access Network. *IEEE Trans. Veh. Technol.* **2015**, *65*, 2063–2077. [CrossRef]
28. Boulos, K.; ElHelou, M.; Lahoud, S. RRH clustering in cloud radio access networks. In Proceedings of the International Conference on Applied Research in Computer Science and Engineering (ICAR), Beiriut, Lebanon, 8–9 October 2015; pp. 1–6.
29. Miyanabe, K.; Suto, K.; Fadlullah, Z.M.; Nishiyama, H.; Kato, N.; Ujikawa, H.; Suzuki, K.I. A cloud radio access network with power over fiber toward 5G networks: QoE-guaranteed design and operation. *IEEE Wireless Commun.* **2015**, *4*, 58–64. [CrossRef]
30. Liu, L.; Yu, W. Cross-Layer Design for Downlink Multihop Cloud Radio Access Networks With Network Coding. *IEEE Trans. Signal Process.* **2017**, *65*, 1728–1740. [CrossRef]
31. Wang, X.; Wang, K.; Wu, S.; Di, S.; Jin, H.; Yang, K.; Ou, S. Dynamic Resource Scheduling in Mobile Edge Cloud with Cloud Radio Access Network. *IEEE Trans. Parallel Distrib. Syst.* **2018**, *29*, 2429–2445. [CrossRef]
32. Budhdev, N.; Maity, A.; Chan, M.C.; Mitra, T. Load balancing for a user-level virtualized 5G cloud-RAN. In *MobiArch '22: 17th ACM Workshop on Mobility in the Evolving Internet Architecture, Proceedings of the ACM MobiCom '22: The 28th Annual International Conference on Mobile Computing and Networking, Sydney, NSW, Australia, 21 October 2022*; Association for Computing Machinery: New York, NY, USA, 2022; pp. 1–6. [CrossRef]
33. Ari, A.A.A.; Gueroui, A.; Titouna, C.; Thiare, O.; Aliouat, Z. Resource allocation scheme for 5G C-RAN: A Swarm Intelligence based approach. *Comput. Netw.* **2021**, *165*. [CrossRef]
34. Zhong, C.-H.; Guo, K.; Zhao, M. Online Sparse Beamforming in C-RAN: A Deep Reinforcement Learning Approach. In Proceedings of the 2021 IEEE Wireless Communications and Networking Conference (WCNC), Nanjing, China, 29 March–1 April 2021; pp. 1–6. [CrossRef]
35. Ye, Y.; Zhang, T.; Yang, L. Joint user association and resource allocation for load balancing in RAN slicing. *Phys. Commun.* **2021**, *49*, 101459. [CrossRef]
36. Suresh, K.; Kumaratharan, N. Call Admission Control Decision Maker Based on Optimized Fuzzy Inference System for 5G Cloud Radio Access Networks. *Wirel. Pers. Commun.* **2021**, *120*, 749–769. [CrossRef]