Vol 6 No 2 (2024) *e*-ISSN: 2682-8383

Journal of Engineering Technology and Applied Physics

Binary Particle Swarm Optimization for Fair User Association in Network Slicing-Enabled Heterogeneous O-RANs

Jing Ren Sue[1], Teong Chee Chuah[1,*] and Ying Loong Lee[2]
[1]*Faculty of Engineering, Multimedia University, Cyberjaya, 63100, Malaysia.*
[2]*Lee Kong Chian Faculty of Engineering and Science, Universiti Tunku Abdul Rahman, Bandar Sungai Long, 43200, Malaysia.*
[*]*Corresponding email*: tcchuah@mmu.edu.my, *ORCiD:* 0000-0002-6285-9481
https://doi.org/10.33093/jetap.2024.6.2.3
Manuscript Received: 18 January 2024, Accepted: 21 February 2024, Published: 15 September 2024

***Abstract* — The Open-Radio Access Network (O-RAN) alliance is leading the evolution of telecommunications towards a greater intelligence, openness, virtualization, and interoperability within mobile networks. The O-RAN standard incorporates of many components the Open-Central Unit (O-CU) and Open-Distributed Unit (O-DU), network slicing and heterogeneous base stations (BS). Together, these innovations have given rise to a three-tiered user association (UA) relationship in a type of network called heterogeneous network (HetNet) with network slicing-enabled. There is an absence of efficient UA schemes for achieving fair resource allocation in such network scenario. Hence, this study formulates the fairness-aware UA problem as a utility-based combinatorial optimization problem, which is computationally hard to solve. Hence, an efficient Binary Particle Swarm Optimization (BPSO)-based UA scheme is proposed to solve the problem. Through simulations of an O-RAN based HetNet with network slicing-enabled, performance of the proposed BPSO-UA scheme is compared against two other baseline UA schemes. Results demonstrate the effectiveness of the proposed BPSO-UA scheme in achieving high fairness through equitable network slicing resource allocation, thereby leading to higher user connectivity rate and comparable average spectral efficiency. This innovative approach sheds light on the potential of metaheuristic algorithms in tackling intricate UA challenges, offering valuable insights for the future design and optimization of mobile networks.**

Keywords*— User association, Heterogeneous network, Open-radio access network, Network slicing, Binary Particle Swarm Optimization.

I. INTRODUCTION

The O-RAN architecture represents a departure from traditional proprietary and monolithic radio access network (RAN) into a disaggregated architecture where the radio units, distributed units, and control units are decoupled for a more modular and interoperable approach [1]. This functional split results in the RAN being split into three segments of backhaul, mid-haul and fronthaul connections. Fronthaul represents the link between the base station (BS) and the processing-centric Open-Distributed Unit (O-DU). The mid-haul acts as an intermediate segment that facilitates communication between O-DUs and the Open-Central Unit (O-CU), which aggregates network traffic from multiple sources before forwarding it to the next network segment. Lastly, the backhaul is responsible for connecting the core network to aggregation points of O-CU which completes the link between the access network to the mobile telecommunication infrastructure. The introduction of granular control and disaggregated function into a traditionally monolithic RAN architecture is related to two well-established network architecture: heterogeneous network (HetNet) and network slicing. HetNet characterizes a telecommunications network that incorporates different tiers of BS, such as macrocells and small cells, strategically addressing the increasing demands for connectivity and capacity [2]. Network slicing enables the creation of logical networks, each with appropriate isolation, resources and optimized topology to serve specific service categories or customers [3].

User Association (UA) is the process by which users in a mobile network establishing connections to serving BSs. The introduction of network slicing and HetNets has given rise to a three-tiered UA relationship, involving user equipment (UE) and virtualized network slices (NSs) provided by different tiers of BSs. Existing UA schemes for traditional networks typically consider optimizing network parameters independently, leading to distributed control implementations [4]. This method of UA results in the underutilization of small base stations and overloading of macro base stations in HetNets [5]. Hence, the multi-UA problem has been studied along with other network functions such as power control and radio resource allocation to enhance load balancing within HetNets.

The authors in [6] have developed a UA scheme that achieves high energy efficiency while ensuring minimum user rates and effectively offloading users to smaller cells. Another study investigated a low-complexity, distributed biasing method that increases the likelihood of users connecting to smaller cells, outperforming conventional UA based on maximum received signal strength [7]. The authors in [8] and [9] focused on heuristic algorithms for UA and resource allocation in HetNets, while others explored deep learning approaches (e.g., Zhao *et al.* [10], Zhang *et al.* [11]). Recent research delved into UA for network slicing-enabled networks. Amine *et al.* [12] proposed a novel network slicing architecture facilitating UA within 5G ultra-dense HetNets, while Ye *et al.* [13] addressed joint UA and resource allocation for load balancing. However, Jayanthi *et al.* [14] introduced an evolutionary approach for UA within a multi-tenant sliced HetNet, albeit under the unrealistic assumption of each network slice being served by a single base station, without accounting for co-channel interference. Joda *et al.* [15] combined UA with placement optimization in O-RAN but did not consider network slicing. Nizam *et al.* [16] addressed UA in hybrid access networks with slicing but adopted a non-O-RAN architecture in their model.

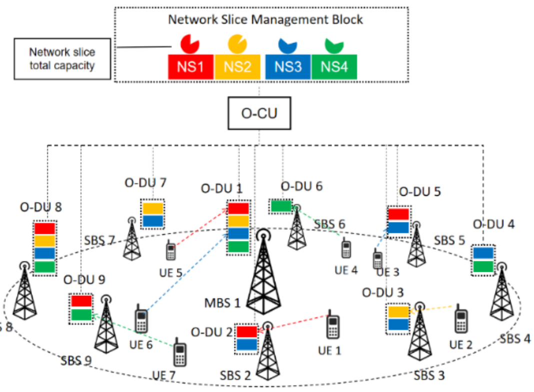
Despite these efforts, there remains a notable research gap concerning UA schemes for network slicing-enabled HetNets deployed under an O-RAN architecture. This gap is particularly critical due to the three-tiered association relationship between UEs, NSs, and BSs. Efficient UA is paramount for achieving load balance among NSs and preventing either overloading or underutilization. Importantly, none of the previously mentioned UA schemes are suitable for addressing this challenge, as they lack the capability to handle three-tiered associations effectively. This challenge arises from the combination of continuous (e.g., signal strength) and discrete (e.g., slice selection) variables, the presence of multiple optimal solutions, and discontinuities within the feasible solution space. Classical optimization techniques, designed for continuous and well-defined problems, prove inadequate in such scenarios.

This research addresses this critical gap by proposing a novel three-tiered UA scheme leveraging the power of Computational Intelligence (CI) [17]. CI techniques excel at tackling complex optimization problems with mixed variable types, multiple optima, and non-smooth solution spaces. We adopt the Binary Particle Swarm Optimization (BPSO) algorithm [18], renowned for its robustness and efficiency in exploring diverse solution landscapes and handling discrete variables. BPSO mimics the collective foraging behaviour of swarms, iteratively refining candidate solutions to reach optimal placements of UEs, NSs, and BSs, considering load balancing and resource utilization across NSs.

The rest of the current paper is structured as follows. In Section II, we introduce the system model and formulate the three-tiered UA problem. Section III presents the proposed BPSO based UA scheme and discusses the process of the BPSO algorithm in detail. Section IV presents the simulation results and discussion. Section V concludes the paper.

II. SYSTEM MODEL AND PROBLEM FORMULATION

The O-RAN architecture adopts a control and user plane separation paradigm, featuring the presence of two distinct entities: The O-DU governing the user plane, and the O-CU managing the control plane. In the context of a UA problem, the O-CU makes a centralized decision about UA and routing based on various factors, including load balancing, network conditions, and user requirements. On the other hand, the O-DU forwards user traffic and consumes the network resources, i.e., NSs. Figure 1 shows the system model of an O-RAN based HetNet with network slicing-enabled.



O-DU Open-Distributed Unit
O-CU Open-Central Unit
MBS Macro Base Station
SBS Small Base Station
UE User Equipment
NS Network Slice

Fig. 1. System model of the UA problem.

From the user plane perspective, the Network Slice Management Block is responsible for overseeing the various types of NSs and their overall capacity. Each BS's O-DU has the capability to accommodate various types of slices, aligning with the network operator's deployment strategy. Despite this flexibility, each slice is consistently identified within the management block based on its overall capacity. The system model delineates multiple Small Base Stations (SBS) within the coverage radius of a single Macro Base Station (MBS). Each BS is paired with an O-DU, which in turns hosts the NSs associated with User Equipment (UEs). However, for a UE to establish an association with a BS, certain conditions must be met: the UE must be subscribed to the corresponding NS, possess sufficient signal strength, and there must be available capacity within the management block for the given slice.

Based on the system model, a mathematical definition is given to the three-tiered UA problem as a maximization of the summation of utility function, $f_\alpha(z)$ of various NSs deployed in the network:

$$\max_{\{a_{kmu}\}} \sum_{m\in M} f_\alpha\left(\frac{\sum_{k\in K}\sum_{u\in U} x_{mk} a_{kmu} R_{ku}}{R_{\max,m}}\right) \quad (1)$$

Firstly, the numbers of BS, NS and UE are represented by the variables *K*, *M* and *U*, while their lowercase counterparts, i.e., *k*, *m*, and *u* denote specific instances of individual BSs, NSs and UEs, respectively. The O-DU which holds the association of a NS *m* to BS *k* is modelled as an independent binary matrix variable x_{mk}. If $x_{mk} = 1$, it means that NS *m* is available at the O-DU belonging to BS *k*, otherwise $x_{mk} = 0$. The variable, a_{kmu} is the 3D UA matrix used to represent the network wide UA association. If $a_{kmu} = 1$, it means that UE *u* associates with NS *m* to BS *k*. R_{ku} denotes the spectral efficiency between BS *k* and UE *u*, serving as a metric to characterize the network channel condition. It quantifies the effectiveness of data transmission over the allocated frequency. $R_{\max,m}$ is the maximum spectral efficiency that a NS *m* can allocate within the management block. The utility function, f_α (z) is defined as follows:

$$f_\alpha(z) = \begin{cases} \dfrac{z^{1-\alpha}}{1-\alpha} & \alpha \neq 1 \\ \log(z) & \alpha = 1 \end{cases} \quad (2)$$

The variable α indicates a fairness notion, where α = 1 corresponds to proportional fairness, α = 2 signifies delay-fairness, and as α → ∞ indicates max-min fairness [19]. Proportional fairness indicates the notion of logarithmic growth that signifies a significant gain in fairness initially. However, as the allocation reaches a certain level, the incremental improvement in fairness diminishes, reflecting a balanced distribution of spectral efficiency across users. As for both the delay and max-min fairness, their formula indicates that as the resulting network-wide resource usage increases the contribution to the overall fairness of the resource allocation is small. The maximization problem is subject to the following constraints:

$$\sum_{k\in K}\sum_{u\in U} x_{km} a_{kmu} R_{ku} \leq R_{\max,m} \ \forall m \in M \quad (3)$$

$$\sum_{m\in M}\sum_{k\in K} x_{km} a_{kmu} R_{ku} \geq R_{\min,u} \ \forall u \in U \quad (4)$$

$$\sum_{m\in M}\sum_{k\in K} x_{km} a_{kmu} \leq 1 \ \forall u \in U \quad (5)$$

$$a_{kmu} \in \{0,1\} \ \forall u \in U, k \in K, m \in M \quad (6)$$

Constraint Eq. (3) guarantees that the total aggregate transmission rate of all the UEs associated with NS *m* does not exceed the maximum aggregate transmission rate of NS *m*. To illustrate, envisioning each NS as a pie, the slices of these pies are distributed among the users subscribed to NS *m* and associated with the network. Consequently, the total pie allocated to the users logically cannot exceed the overall pie designated for NS *m*. Constraint Eq. (4) ensures that each UE *u* associated with BS *k* that can meet the minimum spectral efficiency requirement through NS *m*. Back to the analogy of pie sharing, UEs express a preference for the size of their pie slice; failure to meet this preference implies the UE will not get a slice of pie. Constraint Eq. (5) guarantees that each UE can only associate with one NS via one BS at any given time. In constraint Eq. (6), the variable a_{kmu} takes the value of either zero or one.

III. BINARY PARTICLE SWARM OPTIMIZATION (BPSO)

A. BPSO Working Principle

The BPSO deals with the optimization problem with binary variables, where the potential solutions are represented as binary strings (sequences of 0s and 1s). Each element in the binary string is considered a decision variable which takes either zero or one. Let p_i(t) denote a single position at time step *t*, its new position, $p_i(t+1)$ is expressed as follows,

$$p_i(t+1) = p_i(t) + v_i(t+1) \quad (7)$$

To effectively explore the solution space, PSO requires a mechanism for changing the positions of its particles. The position of the particle for the next iteration, $p_i(t+1)$ is the summation of the current position, $p_i(t)$ with a velocity, $v_{ij}(t+1)$. The changes in the position of the particle facilitate the search for a better solution. The ranking of a particle is determined by its fitness value. The fitness value is computed using a fitness function that maps a particle to a real value. This approach enables the quantification and comparison of a particle's performance. In a maximization problem, the optimization process deems the particle with a higher fitness value to be a better solution. The particle keeps track of its position with the highest fitness value as personal best or *pbest*. The global best or *gbest* particle is the particle with the highest *pbest* value in an iteration. Based on the two information, the velocity at the next time step, $v_{ij}(t+1)$ is defined as

$$\begin{aligned} v_{ij}(t+1) = v_{ij}(t) &+ c_1 r_1 (y_i(t) - p_{ij}(t)) \\ &+ c_2 r_2 (\hat{y}_i(t) - p_{ij}(t)) \end{aligned} \quad (8)$$

The variables c_1 and c_2 are the acceleration constants which determine the exploration and exploitation aspects of the algorithm, respectively, and r_1 and r_2 are random numbers. The variable $y_i(t)$ represents the single *pbest* particle, while $\hat{y}_i(t)$ represents the *gbest* particle. Let $y_i(t)$ represent the *pbest* experience of the particle, and the *pbest* particle of the next iteration is determined as follows,

$$y_i(t+1) = \begin{cases} y_i(t) & \text{if } f(p_i(t+1)) \le f(y_i(t)) \\ p_i(t+1) & \text{if } f(p_i(t+1)) > f(y_i(t)) \end{cases} \tag{9}$$

The fitness of *pbest*, $f(y_i(t))$ is compared to the fitness of the position of the next time step, $f(p_i(t+1))$. Considering a maximization problem, the particle should always hold a higher *pbest*. Hence in the occasion of the latest position having a higher fitness, the particle stores the new position, $p_i(t+1)$ as its *pbest*. Then, the *gbest* particle, $\hat{y}_i(t)$ is determined through finding the largest value *pbest* particle of that iteration.

$$\hat{y}_i(t) = \max\{y_1(t), \ldots, y_i(t)\} \tag{10}$$

The term exploration and exploitation in BPSO arises from the degree of influence exerted by the *pbest* and *gbest* particles on the results of subsequent generation. When the velocity is primarily guided by the particle's own *pbest*, the BPSO is considered as explorative, which emphasizes a particle’s own exploration of the solution space. However, if the velocity is predominantly driven by the *gbest* of the entire swarm, the BPSO is considered to be exploitative, focusing on refining and exploiting the current best solution found by the swarm. Striking the right balance between exploration and exploitation is a key aspect of PSO to achieve an effective and efficient search for optimal solutions.

The PSO algorithm was originally designed to operate in continuous space. A discrete binary version of PSO was also introduced [18]. In a continuous space, the particle’s position assumed continuous values and is changed by adding a continuous velocity. For a discrete PSO, the particle’s position is a vector that holds binary value of ‘0’ or ‘1’. For a single bit in the vector to change, it is determined by its continuous-valued probability. This probability is the velocity normalized to a value between 0 and 1 using the sigmoid function.

$$v_{sigmoid}(t) = \frac{1}{1+e^{-v(t)}} \tag{11}$$

By confining the velocity within the range of [0,1], it becomes interpretable as the probability of a single bit within the particle changing. Based on the probability, the position in the next time step, $p_{ij}(t+1)$ is updated as follows,

$$p_{ij}(t+1) = \begin{cases} 1 & rand \le v_{sigmoid}(t+1) \\ 0 & otherwise \end{cases} \tag{12}$$

In order to determine the position of the particle in the next time step, $p_{ij}(t+1)$, a random value is generated to compare to the probability. If the generated random value is lower than the probability threshold, the corresponding bit in the particle will be set to '1'. Figure 2 illustrates the 2D matrixes depicting the relationship between velocity and particle position.

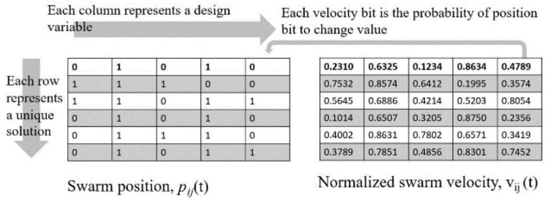


Fig. 2. Relationship between position, $p_{ij}(t)$ and velocity, $v_{ij}(t)$ in BPSO.

The BPSO algorithm maintains a 2D matrix of swarm of particles with each row representing a particle’s unique solution to the problem. The velocity, traditionally viewed as the rate of movement in continuous space, is redefined in BPSO as the probability of changing a single bit within a particle. To represent velocity as a probability, normalization is required, achieved by applying the sigmoid function to the velocity values. After which the probability still depends on the particle’s *pbest* and the *gbest* particle at iteration *t* to experience changes and to guide the solution. The *pbest* and *gbest* particles are selected by ranking the particles and choosing them based on the fitness value calculated by the fitness function. The whole process is repeated until the maximum number of iterations is reached or the improvement on the fitness function value is negligible.

B. *Proposed BPSO Based UA Scheme*

For the UA scheme to leverage on the robustness and computation capabilities of the BPSO algorithm, it is imperative to establish a representation of the network wide UA. The 3D UA matrix, a_{kmu} consists of *K* rows, *M* columns and *U* pages, representing the BS, NS and UE, respectively. The elements of a_{kmu} can take binary values, either 0 or 1. A value of 1 indicates that user *u* is associated with the network through base station in row *k* and the network slice in column *m*. This method of formulating the UA matrix sets the foundation for constructing a solution space for the algorithm. Figure 3 illustrates the relationship between the HetNet with network slicing-enabled and the UA matrix.

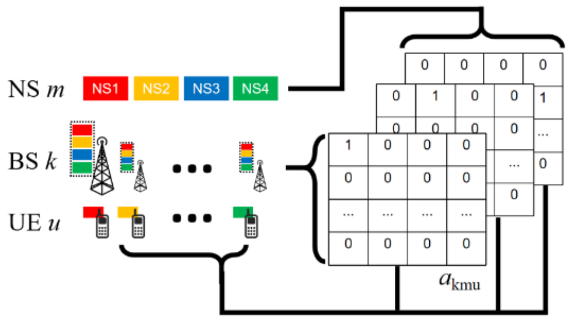


Fig. 3. Generation of the user association matrix a_{kmu} from the system model.

The linkage between the UA matrix to the BPSO is built by a matrix transformation operation of flattening the 3D matrix into a single dimensional vector while preserving its order. Figure 4 shows how the relationship is drawn between the 3D UA matrix and the BPSO particle.

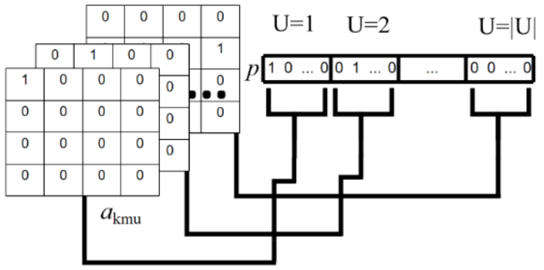


Fig. 3. Mapping the 3D UA matrix, a_{kmu}, into a single vector of BPSO particle, *p*.

The flattening of the 3D matrix forms a single vector, where network wide UA is representable by index of the particle. Figure 5 illustrates the internal details of a single BPSO particle, denoted as *p*, and its correlation with indices corresponding to BSs and NSs.

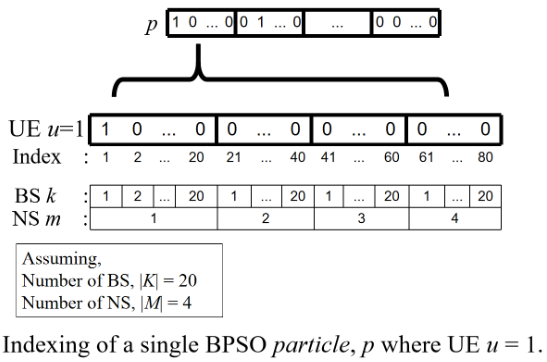


Fig. 4. Indexing of a single BPSO particle *p* for UE with user index *u* = 1.

The linearization of a_{kmu} into a single vector of a particle, *p* streamlines the optimization process for the implementation of BPSO algorithmic process. This reduction of the dimensionality of the UA matrix enhances the efficiency of swarm-based optimization techniques like BPSO. The linearized vector allows for improved navigation of the solution space, leading to enhanced computational efficiency and compatibility with optimization algorithms.

A UA scheme in the context of wireless communication networks involves the assignment of UEs to specific network entities, such as BSs or access points. The BPSO based UA scheme represents an innovative approach to associating users to the network in the context of the UA problem defined in our system model. Unlike conventional methods, the proposed scheme introduces a heuristic optimization algorithm that treats the user association problem as a search process in a solution space represented by particles. In this section, the linkage of the UA scheme and how the equations in the BPSO algorithm is applied is described. Figure 6 presents the workflow of BPSO for generating a feasible solution.

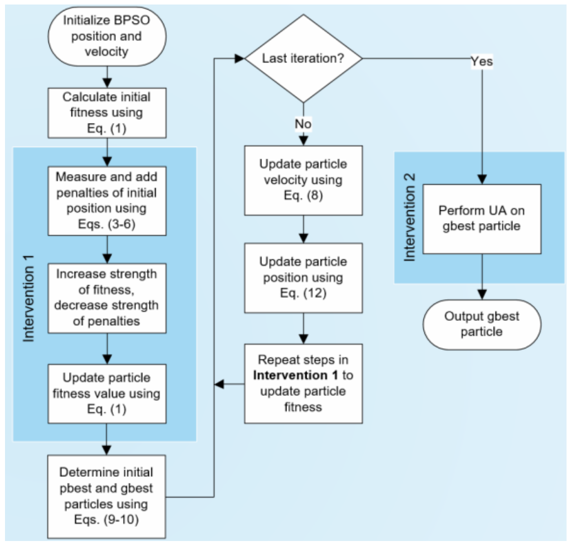


Fig. 5. Flowchart of the BPSO based UA scheme.

There are mainly two phases for the BPSO based UA scheme: an initialization phase and an iterative phase. During this phase, several key parameters, including the particle's velocity, position, fitness, *pbest*, and *gbest* particle, are initialized with predetermined values. An intervention is introduced to this initialization phase in the form of measuring and adding the penalties, adjusting the strength of fitness and penalties, and storing them as the new fitness to the particles. After the initialization process, the iteration process will start by updating the velocity, position, calculating fitness with the same mentioned intervention and updating the *pbest* and *gbest* particle. At the end of the iteration process, the *gbest* solution undergoes a final refinement step before it is presented as the ultimate solution.

After initializing the position and velocity of the BPSO swarm, the fitness of the initial position is calculated. The *pbest* of the particles are initialized to their initial fitness, the initial *gbest* is set to be the best initial *pbest*. The iteration process starts with updating the velocity according to the velocity update rule in Eq. (8). The change of velocity is influenced by the *pbest* and *gbest* particle. By using the normalized velocity, the position of the particle is updated based on the position update rule in Eq. (12). The fitness and the penalties of the new position are calculated and stored as the particle's fitness. The *pbest* update rule in Eq. (9) is applied to determine whether to assign the *pbest* to the newly updated position, whereas Eq. (10) is applied to determine the *gbest* particle.

The transition to an unconstrained maximization problem through penalty method functions introduces certain challenges, notably impeding particle exploration due to frequent penalty function violations. Integrating the penalty function into the objective function results in fitness values falling within a range of significantly large negative numbers, posing challenges for convergence. To address these issues and enhance the search for the fairest UA solution, two strategic interventions are introduced into the BPSO algorithm. These interventions aim to refine the optimization process and overcome obstacles posed by penalty functions, making the pursuit of optimal UA solutions more effective and nuanced.

In Intervention 1, constraints on the objective functions are incorporated as penalties within the fitness function. These penalties, when added to the objective functions, result in a significant negative value of the objective function. What proved to be highly undesirable was the persistent stagnation of the *gbest* fitness throughput the entire iteration. To address this issue, a work around is employed, whereby the strength of the fitness is linearly increased while the strength of the penalties is linearly decreased. The implementation of this approach has significantly enhanced particle exploration and yielded more favourable results. The immediate implication of the intervention is the convergent behaviour of the fitness of the *gbest* particle. Conventionally, the objective function is a graph that shows the quantified performance of the solution in an optimizing algorithm. In this case, the intervention made to the objective function is observed to have facilitated the quality of the solution provided by the *gbest* particle.

Intervention 2 occurs at the end of the iteration process, where the particle is inspected to ensure compliance with the constraints and user associations within the network. While the first intervention may have produced particles that adhere to the constraints imposed as penalties, the solution at this stage remains unsatisfactory. The second intervention is simply a process of user association based on the remaining spectrum efficiency in the NSs. Several parameters for the proposed BPSO UA scheme are defined during the initialization phase before the iterative heuristic search process is initiated. Table I shows the values and description of these parameters.

In Eq. (8), the stochastic influence on the velocity update rule is determined by constants c_1, c_2, and random numbers r_1 and r_2. The constant c_1 signifies exploration, influencing the particle based on its own past position, while c_2 represents exploitation, with the particle influenced more by the global best particle. Varying these coefficients leads to different algorithmic behaviours (see Section 16.4 of [17]).

For consistent influence, static and equal values for c_1 and c_2 attract particles toward the average solution between their personal best (*pbest*) and the global best (*gbest*). Choosing a value of 1.5 for both c_1 and c_2, which can be determined based on the constriction factor approach in [20] would ensure smoother particle exploration and exploitation trajectories. A larger swarm size is beneficial for better exploration in the search space, but it comes with increased computational time. Here, a swarm size of 50 particles is chosen, which generally performs well for a PSO algorithm [21]. With the aforementioned settings, we have experimentally assessed the convergence performance of the proposed algorithm based on a simulation scenario with settings given in the next section, and found that the proposed algorithm converges within 100 iterations. As such, a maximum of 100 iterations is chosen for the proposed algorithm. It is noteworthy that we do not use the convergence criterion, where the BPSO terminates when the fitness function value can no longer be improved. This is because in some situations, the proposed algorithm could continue to iterate for a larger number of iterations, which is impractical for the real-time UA process. That said, fixing the maximum number of iterations to 100 also does not always guarantee convergence, which is a possible limitation. Nevertheless, in the next section, we show that the proposed algorithm still performs well with 100 iterations. The α value determines the function used in calculating the objective function (refer to Eq. (2)). Setting α to 1 employs a logarithmic function in fitness calculation, indicating diminishing returns for higher spectral efficiency sums. This concept, borrowed from packet scheduling and flow control, aligns with limiting user rates to prevent congestion and promote overall network fairness.

Table I. Parameter definitions for the initialization phase of BPSO.

Parameter	Value	Description
c_1	1.5	Refer to Eq. (8).
c_2	1.5	Refer to Eq. (8).
Number of particles	50	The number of unique UA solutions.
Maximum iteration	100	The maximum number of iterations for BPSO.
alpha	1	Refer to Eq. (2).
Particle's velocity	$\{\mathbb{R}+,\ldots\}$	The velocity is initialized to random positive real value.
Particle's position	$\{0_2,1_2,\ldots\}$	The position is a vector of binary numbers representing a unique solution.
Particle's fitness	$\mathbb{R}$	A positive real number quantifying the performance of a position.
Particle's *pbest*	$\{0_2,1_2,\ldots\}$	Holds the particle's own position with the highest fitness.
Particle's *gbest*	$\{0_2,1_2,\ldots\}$	Holds the best particle among all *pbest* particles.

IV. RESULTS AND DISCUSSION

Several parameters and equations collectively define the network channel for the HetNet model, offering insights into the scale, configuration, and capabilities of the simulated O-RAN HetNet with network slicing-enabled. Firstly, the log-distance path loss model is employed to characterize signal strength to each UE from the BSs. The specific path loss equations differ depending on whether it is an MBS or an SBS. For the MBS, the path loss equation is 140.7 + 36.7 log (d), while for a SBS, it is 128.1+37.6 log (d). The Shannon's capacity, $C = B \log_2 (1 + \text{SINR})$, where C represents capacity, B signifies bandwidth, is employed to quantify the quality of the received signal. The signal-to-interference-plus-noise ratio (SINR) is formulated as $\text{SINR} = \frac{P_{\max,k} G_{ku}}{\sum_{i \in K\{k\}} P_{max,i} G_{iu} + N_0}$, where $P_{\max,k}$ is the maximum transmit power of BS k, G_{ku} is the channel gain between BS k and UE u, and N_0 represents the noise. The summation term, i.e. $\sum_{i \in K\{k\}} P_{max,i} G_{iu}$ considers the transmit power of BS i, which represents the rest of the BSs other than BS k, as interference power. By calculating the ratio of Shannon's capacity to available bandwidth, B, we obtain $R = \log_2(1+\text{SINR})$, which quantifies how efficiently information is conveyed within the available spectrum. Hence in the UA problem, the value of R_{ku}, is the radio link capacity between a BS k to a UE u is considered in the system model. Next, the simulation considers a total of 20 BSs, comprising one MBS and 19 SBS. Four distinct NSs are defined. The UEs are categorized into four groups with varying sizes: 50, 100, 150, and 200. The maximum capacity for each of the four NSs is defined by $R_{\max}$, with values {10, 20, 25, 30} in bits per second per Hz. The simulation assumes a MBS radius of 500 meters. Additionally, the maximum transmit power of the MBS and maximum transmit power of SBSs are assumed to be 40 dBm and 30 dBm, respectively.

The following presents performance results collected from the proposed UA scheme using BPSO and two other baseline UA schemes: BSNS (Baseline Scheme 1), where UEs are initially associated with BSs based on the maximum SINR. Subsequently, UEs are associated with NSs available in the associated BSs. The second scheme, NSBS (Baseline Scheme 2), prioritizes associating UEs with NSs capable of meeting the target data rates. This is followed by the association of UEs with BSs where the associated NSs can be accessed, based on maximum SINR. The proposed BPSO-UA scheme simultaneously evaluates multiple instances of the UA matrix, ranks them by their objective function and selects the best solution as the UA solution for the network. This approach employs heuristics to identify the best UA matrix that satisfies the network's performance metrics.

$$J = \frac{\left(\sum_{m=1}^{|M|} \frac{R_m^{'}}{R_{\max,m}} \right)^2}{|M| \sum_{m=1}^{|M|} \left(\frac{R_m^{'}}{R_{\max,m}} \right)^2} \tag{13}$$

The first performance metrics is Jain's Fairness Index (FI), which quantifies the fairness of resource distribution across multiple NSs. Equation (13) provides the mathematical definition for Jain's FI, denoted as J.

The variable R_m' denotes the spectral efficiency of a user connected to NS m using a specific UA scheme, while $R_{\max}$ represents the total network slice capacity. Equation (13) quantifies the resource dispersion across NSs concerning their allocated capacity. Jain's FI yields a continuous value within the range of [0, 1], where $J = 0$ corresponds to the least fair allocation and $J = 1$ represents the fairest allocation, ensuring that all entities received equal benefits. Utilizing the FI value, researchers can calculate the average variation or dispersion in the allocated spectral efficiency of each slice, offering insights into how network slice efficiency varies across the three UA schemes. Figure 7 shows the result of Jain's FI for the different schemes, considering four user groups.

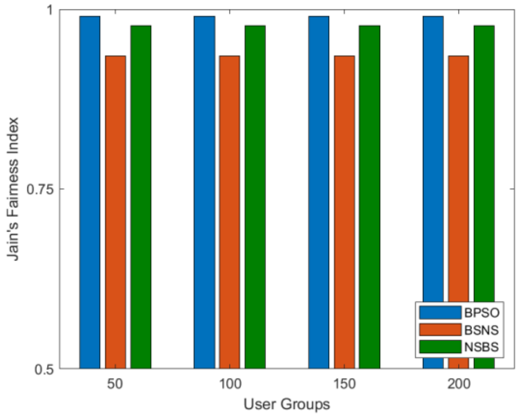


Fig. 6. Jain's fairness index for the three UA schemes in four user groups.

The analysis across four user groups (i.e., 50, 100, 150, and 200 users) reveals consistent results in terms of FI values for the three UA schemes. Specifically, the BSNS scheme lags behind both NSBS and the proposed BPSO-UA scheme. The results indicate that out of the three schemes, the proposed scheme achieves superior fairness in the distribution of NSs. Nevertheless, it is important to note that fairness within NS distribution primarily reflects operator-side equity in resource allocation, ensuring a well-balanced network resource load. For a comprehensive evaluation of fairness from the user's perspective, we present the corresponding fairness metric in Eq. (14), referred to as the user connectivity rate.

$$\text{User connectivity rate}\,(\%) = \frac{U'}{|U|} \times 100 \tag{14}$$

The user connectivity rate indicates the percentage of satisfied users, U' in the network relative to the total user population, $|U|$. Satisfied users are those whose minimum data speed requirements are met. Users unable to meet their needs are dropped or left unconnected to the network. The network adheres to a policy defining the available throughput to be shared among users, akin to an aggregate shared throughput. When users connect to the network, they consume a portion of this shared throughput, and a surge in connections depletes the shared throughput more rapidly. Consequently, the network's goal is to establish connections for as many users as possible, while ensuring that they receive, at a minimum, the required throughput for effective network usage. Figure 8 shows the user connectivity rate for all four user groups across the three UA schemes.

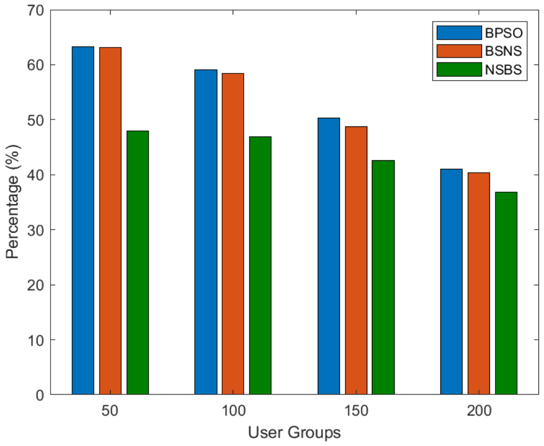


Fig. 7. User connectivity rate for the three UA schemes in four user groups.

While NSBS can more fairly distribute the load of the network resources, the BSNS scheme facilitates a higher number of connected users. The proposed BPSO-UA scheme demonstrates a modest improvement in this performance metric compared to the other two schemes. A higher user connectivity rate implies that the BPSO scheme can accommodate more users without necessitating disconnections. However, in the worst-case scenario of channel conditions, involving noise and interference from all BSs, a significant portion of users may still experience insufficient signal strength. Consequently, it becomes imperative to assess the quality of user connections by quantifying their average spectral efficiency, as described by Eq. (15).

$$\text{Average spectral efficiency} = \frac{\sum_{k=1}^{|K|}\sum_{u'=1}^{|U'|} R'_{ku}}{|U|} \tag{15}$$

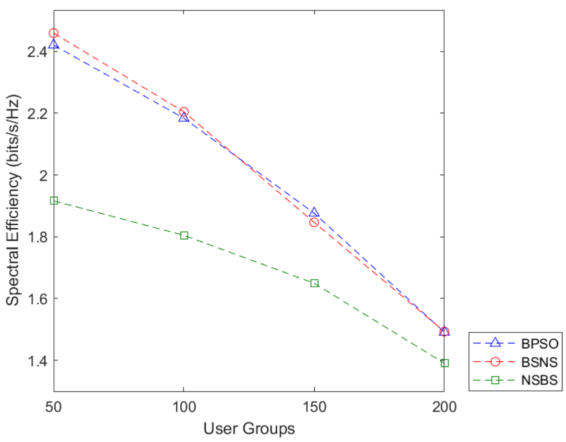


Fig. 8. Average spectral efficiency for the three UA scheme in four user groups.

The average spectral efficiency is calculated by summing the spectral efficiency achieved by the connected users, represented as R_{ku}', and then dividing this sum by the total number of users in the network, $|U|$. As illustrated in Fig. 9, due to resource limitation, it is evident that as the network's user population grows, the average spectral efficiency of each user follows a decreasing trend.

The BSNS scheme serves as a benchmark by associating users with the network according to maximum received signal strength, representing an upper limit for per-user spectral efficiency. The proposed BPSO approach achieves a similar average spectral efficiency per user as BSNS. The primary objective of BPSO is to maximize the total spectral efficiency of the network. While there may be a slight reduction in per-user spectral efficiency with a smaller number of users, the proposed BPSO scheme demonstrates the potential for achieving comparable or even higher average spectral efficiency in scenarios involving larger user groups.

V. CONCLUSION

In this paper, we address the complex challenge of three-tiered UA in a network slicing-enabled heterogeneous O-RAN, involving UE connections to NSs provided by various BS tiers. We formulate this challenge as a utility-based combinatorial optimization problem, which presents significant computational difficulties for conventional approaches. To address this, we leverage the BPSO metaheuristic algorithm. Results show that the proposed BPSO UA scheme surpasses two baseline schemes in terms of fairness, user connectivity rate, and spectral efficiency. For future work, we suggest exploring the scalability and adaptability of the BPSO algorithm in larger, dynamic network scenarios.

ACKNOWLEDGEMENT

This work was supported in part by the Ministry of Higher Education Malaysia under Grant FRGS/1/2019/ICT05/MMU/01/1.

REFERENCES

[1] O-RAN WG4, "O-RAN Fronthaul Control, User and Synchronization Plane Specification v07.02," 2022.

[2] Y. L. Lee, T. C. Chuah, J. Loo and A. Vinel, "Recent Advances in Radio Resource Management for Heterogeneous LTE/LTE-A Networks," *IEEE Commun. Surv. and Tut.*, vol. 16, no. 4, pp. 2142–2180, 2014.

[3] 3GPP TS Group, "5G; Management and Orchestration; Concepts, Use Cases and Requirements," ETSI TS 128 530 V17.2.0 (2022-05), 2022.

[4] E. Hossain, M. Rasti, H. Tabassum and A. Abdelnasser, "Evolution Toward 5G Multi-Tier Cellular Wireless Networks: An Interference Management Perspective," *IEEE Trans. Wirel. Commun.*, vol. 21, no. 3, pp. 118–127, 2014.

[5] A. Khandekar, N. Bhushan, J. Tingfang, and V. Vanghi, "LTE-Advanced: Heterogeneous Networks," in *2010 Europ. Wirel. Conf.*, pp. 978–982, 2010.

[6] H. Pervaiz, L. Musavian and Q. Ni, "Joint User association and Energy-Efficient Resource Allocation with Minimum-Rate Constraints in Two-Tier HetNets," in *IEEE Int. Symp. Person., Indoor and Mobile Radio Commun.*, pp. 1634–1639, 2013.

[7] Q. Ye, B. Rong, Y. Chen, M. Al-Shalash, C. Caramanis and J. G. Andrews, "User Association for Load Balancing in

Heterogeneous Cellular Networks," *IEEE Trans. Wirel. Commun.*, vol. 12, no. 6, pp. 2706–2716, 2013.

[8] A. Mesodiakaki, F. Adelantado, L. Alonso and C. Verikoukis, "Energy-Efficient Context-Aware User Association for Outdoor Small Cell Heterogeneous Networks," in *2014 IEEE Int. Conf. Commun.*, pp. 1614–1619, 2014.

[9] W. K. Lai and J. K. Liu, "Cell Selection and Resource Allocation in LTE Advanced Heterogeneous Networks," *IEEE Access*, vol. 6, pp. 72978-72991, 2018.

[10] N. Zhao, Y.-C. Liang, D. Niyato, Y. Pei, M. Wu and Y. Jiang, "Deep Reinforcement Learning for User Association and Resource Allocation in Heterogeneous Cellular Networks," *IEEE Trans. Wirel. Commun.*, vol. 18, no. 11, pp. 5141-5152, 2019.

[11] Y. Zhang, L. Xiong and J. Yu, "Deep Learning Based User Association in Heterogeneous Wireless Networks," *IEEE Access*, vol. 8, pp. 197439-197447, 2020.

[12] M. Amine, A. Kobbane and J. Ben-Othman, "New Network Slicing Scheme for UE Association Solution in 5G Ultra Dense HetNets," in *2020 IEEE Int. Conf. Commun.*, Dublin, pp. 1-6, 2020.

[13] Y. Ye, T. Zhang and L. Yang, "Joint User Association and Resource Allocation for Load Balancing in RAN Slicing," *Phys. Commun*., vol. 49, no. 101459, pp. 1-12, 2021.

[14] S. S. Jayanthi, Y. L. Lee and Y. C. Chang, "User Association for Multi-Tenant Heterogeneous Network Slicing Using Genetic Algorithm," in *8th Int. Conf. Comp. and Commun. Eng.*, Kuala Lumpur, pp. 326-330, 2021.

[15] R. Joda, T. Pamuklu, P. E. Iturria-Rivera and M. Erol-Kantarci, "Deep Reinforcement Learning-Based Joint User Association and CU–DU Placement in O-RAN," *IEEE Trans. Netw. and Serv. Manag.*, vol. 19, no. 4, pp. 4097-4110, 2022.

[16] F. Nizam, T. C. Chuah and Y. L. Lee, "User Association for Network Slicing-Enabled Heterogeneous Hybrid Wireless-Wireline Access Networks," in *Int. Conf. Cyber Manag. and Eng.*, Bangkok, 2023.

[17] A. P. Engelbrecht, *Computational Intelligence An Introduction*, Second Edi., John Wiley & Sons, 2007.

[18] J. Kennedy and R. C. Eberhart, "A Discrete Binary Version of the Particle Swarm Algorithm," in *IEEE Int. Conf. Sys., Man, and Cybern.., Computat. Cybern. and Simul.*, pp. 4–8, 1997.

[19] J. Mo and J. Walrand, "Fair End-to-end Window-based Congestion Control," *IEEE/ACM Trans. Netw.*, vol. 8, no. 5, pp. 556–567, 2000.

[20] R. C. Eberhart and Y. Shi, "Comparing Inertia Weights and Constriction Factors in Particle Swarm Optimization," in *2000 Congr. Evolut. Computat. CEC00 (Cat. No.00TH8512)*, La Jolla, vol. 1, pp. 84-88, 2000.

[21] A. P. Piotrowski, J. J., Napiorwski and A. E. Piotrowska, "Population size in Particle Swarm Optimization," *Swarm and Evolut. Computat.*, vol. 58, no. 100718, pp. 1 – 18, 2020.