$PIRS^3A$: A Low Complexity Multi-knapsack-based Approach for User Association and Resource Allocation in HetNets

Bharat Agarwal[1], Mohammed Amine Togou[1], Marco Ruffini[2], and Gabriel-Miro Muntean[1]

[1]School of Electronic Engineering, Dublin City University, Ireland
[2]Computer Science and Statistic Department, Trinity College Dublin, Ireland
[1]bharat.agarwal2@mail.dcu.ie, mohammedamine.togou@dcu.ie, gabriel.muntean@dcu.ie
[2]marco.ruffini@scss.tcd.ie

***Abstract*—The recent worldwide sanitary pandemic has made it clear that changes in user traffic patterns can create load balancing issues in networks (*e.g.,* new peak hours of usage have been observed, especially in suburban residential areas). Such patterns need to be accommodated, often with reliable service quality. Although several studies have examined the user association and resource allocation (UA-RA) issue, there is still no optimal strategy to address such a problem with low complexity while reducing the time overhead. To this end, we propose Performance-Improved Reduced Search Space Simulated Annealing ($PIRS^3A$), an algorithm for solving UA-RA problems in Heterogeneous Networks (HetNets). First, the UA-RA problem is formulated as a multiple 0/1 knapsack problem (MKP) with constraints on the maximum capacity of the base stations (BS) along with the transport block size (TBS) index. Second, the proposed $PIRS^3A$ is used to solve the formulated MKP. Simulation results show that $PIRS^3A$ outperforms existing schemes in terms of variability and Quality of Service (QoS), including throughput, packet loss ratio (PLR), delay, and jitter. Simulation results also show that $PIRS^3A$ generates solutions that are very close to the optimal solution compared to the default simulated annealing (DSA) algorithm.**

***Index Terms*—Heterogeneous Networks, Traffic Optimization, Simulated Annealing, QoS, TBS index, COVID-19.**

I. INTRODUCTION

The coronavirus disease (COVID-19) has caused an unprecedented number of individuals worldwide to work from home. As new digital patterns are emerging, the communications service providers have a critical role in supporting an active community with good quality digital communications [1]. One of the most promising approaches to fulfill this role is the 5G support for HetNets environments. It involves enriching the current cellular network with several smaller and simpler base stations (BS) with broadly varying transmission capacities, coverage areas, carrier frequencies, types of backhaul connections, and communication protocols. For instance, the integration of femtocell base stations (FBSs) with macro-cell base stations (MBSs) enables HetNets to support good quality of service (QoS) when serving diverse users [2], [3].

Among the significant challenges in HetNets are user association (UA) to BS and resource allocation (RA) during service. As a result, the dual UA-RA problem needs to be examined to enable support for high QoS while considering variables such as BS capacity, user requirements, and channel quality. The problem of UA-RA in HetNets has been studied recently. The approaches mainly differ in terms of architecture (*i.e.,* centralized or distributed), the number of parameters considered and the execution time. For instance, Alnoman *et al.* [4] proposed a joint UA-RA de-centralized approach to maximize the overall network throughput using a Mamdani-type fuzzy logic controller (FLC). The users were first classified based on their data rate requirements and the controller decides the amount of bandwidth to allocate to each class.

Wang *et al.* [5] divided the UA-RA problem into two subproblems. The first sub-problem was solved using graph theory by fixing the power allocation (PA), UA, and RA while the second was solved using a difference convex function in which UA-RA was fixed, and PA was solved. Feng *et al.* [6] proposed two schemes for the joint UA-RA problem, one centralized and one distributed. The centralized iterative scheme was broken down into two sub-problems: first, using a cutting plane approach, the UA problem was solved; second, a primary decomposition approach solved the joint frame design and the RA problem. Both sub-problems were iteratively solved to find an optimal solution. The de-centralized scheme used repeated games between users which has shown to achieve the Nash equilibrium. Using the Stackelberg game method, Zhong *et al.* in [7] solved the UA-RA problem while considering the back-haul potential of BSs. Sapountzis *et al.* [8] provided an iterative optimal solution for UA in back-haul restricted HetNets by finding the optimum cell with which to be associated. Luo *et al.* [9] suggested a joint UA-RA scheme to minimize the network packet delay using different QoS-aware UA (QoSA) strategies: descent of block-coordinate, multiplier alternating-direction method, and multi-flow. These algorithms minimized the packet delays in a distributed way at a lower complexity compared to the conventional UA strategies. Barbosa *et al.* [10] proposed the use of DoE (Design of Expert) [11], RSM (Reduced Surface Methodology), and racing algorithms to improve the genetic algorithm (GA) and

Simulated Annealing (SA) efficiency to solve the problems of classical optimization. We have used this work as the base for tuning SA's significant hyper-parameters for solving the multiple 0/1 knapsack problem (MKP). RSM[1] is suggested as a fine-tuning technique by the authors of [12] to achieve greater proximity of regions with promising settings. The racing concept was studied in [13], [14] using F-race, a racing algorithm where candidate configurations are removed using Friedman statistics.

Despite their good performance, most of the aforementioned approaches suffer various limitations. For instance, the approaches in [4] and [5] did not consider the channel quality when solving the UA-RA problem. The schemes in [6] only consider constraints on the wireless back-haul and incur a large overhead, making them unfeasible for large-scale networks.

The limitations of the previous works have motivated us to propose **the Performance Improved Reduced Search Space SA** ($PIRS^3A$), a decentralized solution to the UA-RA problem, formulated as a MKP, considering the maximum BS capacity and the transport block size (TBS) index restrictions. The knapsacks are represented by BSs (MBS and FBS) in this solution, and the items to fit into the knapsacks are represented by instances of user equipment (UE). The item weights are user demands, while item values are the available throughput for each knapsack. Knapsack capacity is the maximum capacity available. We use SA to solve the MKP problem and based on the resource block (RB) utilization rate along with TBS index, the available throughput is determined. To the best of the authors' knowledge, no work has proposed a low complexity solution to the joint UA-RA problem in HetNets while considering several variables.

The rest of this paper is organized as follows. Section II presents the system model. Section III describes the relevant algorithm and presents the algorithm's fine-tuning in terms of parameters range and solution search space. Section IV describes the simulation setup and results. Finally, Section V concludes the paper.

II. System Model

Consider the downlink of a HetNets consisting of fixed BSs and randomly placed UEs, as depicted in Fig. 1. There are two sets of BSs: the set of Macro BS $|S_M| = M_m$ and the set of femtocell BS $|S_f| = M_f$. As the coverage area of these BSs may overlap, we assume that each UE can be within the range of at least one BS. The set of all BSs is denoted as $BS = \{S_M \cup S_f\}$ with $|BS| = M_m + M_f$. A high-speed backhaul with minimal delay (such as high-speed fiber) is connected to all BSs. Let N be the set of UEs located inside the region G and $\psi_j \in \Psi$ be the requested downlink rate (bits per second) of UE j, where Ψ is the discrete set of service classes. In this paper, we are interested in the video service as it requires high bandwidth. Each UE can only be associated with at most one BS at any time instance. We define $\tilde{\mu}$ as the total path loss

[1]The RSM framework is available in DoE software, and a more detailed explanation about its use can be found in [20].

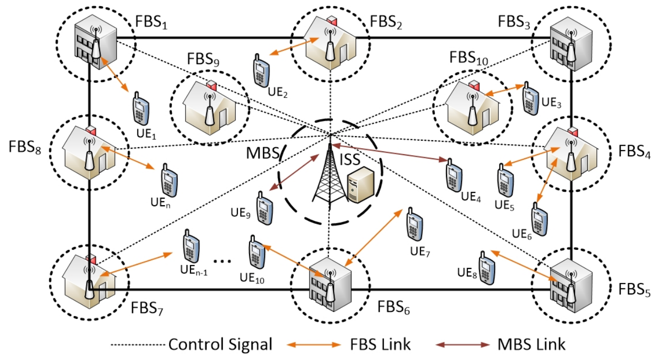

Fig. 1. An example of a two-tier HetNet including one MBS: M_1, and 10 FBSs: $M_2, M_3, \cdots, M_{11}$. The FBSs are usually located at commercial and residential buildings that constitute hotspots for wireless traffic. The UEs $UE_1, UE_2, \cdots, UE_N$ in a region G are either served by either MBS or FBSs selected by Information Service Server (ISS).

(*i.e.,* follows a log-distance path loss model) between BS i and UE j in decibels (dB). Other notations are given in TABLE I.

UEs might not receive high data rates because of time-variable fading channels in wireless communication (i.e., LTE, 5G). This issue is more pronounced in urban areas. In today's wireless communication system, a decentralized scheme is typically used to solve such problem, enabling UEs to communicate with the BSs that provide them with the best channel conditions and which can satisfy their minimum QoS requirements.

There are two actors with distinct perspectives and goals in our scheme, namely, UEs and BSs. On the one hand, each UE wants to achieve the maximum data rate given its QoS requirements. On the other hand, BSs want to fulfill UEs' QoS requirements considering their constraints. Therefore, under the maximum capacity of BSs and TBS index constraint, we describe the objective function as the amount of available throughput (i.e., calculated based on RB usage rate) while considering the entities' perspective and goals.

$$minimize : -f(x) = \sum_{j=1}^{n} p_j x_j \tag{1}$$

subject to

$$\sum_{j=1}^{n} r_{ij} x_j \leq b_i, i \in \{M_m \cup M_f\} \tag{2}$$

$$x_j \in \{0, 1\}, j \in N = \{1, ..., n\} \tag{3}$$

$$q_{ij} \geq \Gamma, i \in \{M_m \cup M_f\}, j \in N = \{1, 2, ..., n\} \tag{4}$$

$$r_{ij} \leq b_i, i \in \{M_m \cup M_f\}, j \in N = \{1, ..., n\} \tag{5}$$

Eq. (1) represents a *hyper-plane*, hence it is a convex function with convex constraints. Eq. (2) implies that the allocation of resources to UEs should not exceed the *maximum BS capacity* (b_i). Eq. (3) shows the *Unique association* property, where UE j can only be associated with one BS at any moment. Eq. (4) implies that each UE j should have a TBS index above a certain threshold Γ to participate in the UA-RA problem (*i.e., TBS index vector*). Finally, Eq. (5) indicates that each BS can serve at least one UE. This MKP problem is formulated as the

TABLE I
TABLE OF NOTATIONS

Notation	Description
N, $\|N\|$, j	set, cardinality, and indexes of UEs
BS, $\|BS\|$, i	set, cardinality and indexes of BSs
$\tilde{\mu}$	path loss between BS i and UE j
ψ_j	requested downlink rates of UE j
b_i	maximum capacity of BS or knapsack i
r_{ij}	resource consumption allocated to each item or UE j for each BS or knapsack i
x_j^i, x	user association decision variable/vector
p_j	profit or available throughput for UE j
Q_i	vector of UEs having TBS index greater than Γ with BS i
U_p	Set of participating UEs

mixed-integer linear program (MILP) which is solved using $PIRS^3A$, described in Section III, to achieve a near-optimal solution and achieve high QoS levels.

III. $PIRS^3A$ ALGORITHM

A. Simulated Annealing (SA) - Overview

SA is a local search algorithm that can circumvent the local optima problem. Over the previous decades, its ease of implementation along with its convergence properties made it a standard algorithm for solving combinatorial optimization problems like MKP. It was named as such because of its similarity to the physical solid annealing process [10], which involves heating and controlled cooling of material by varying temperature. If the temperature decreases very slowly, a stable state can be observed, which cannot be reached if the temperature falls quickly. SA is based on the Metropolis acceptance criterion (MAC) that is based on the Random Walk Metropolis Algorithm (RWMA), which specifies a way of doing dependent sampling from posterior space. It is the simplest version of I^{st} order Markov Chain Monte Carlo (MCMC) algorithms because the decision to the next step in the parameter space only depends on the current state. SA tries to evade the local optima by allowing temporal deterioration of actual solutions (*i.e.,* moves to a solution that corresponds to a worse objective function value), where the deterioration is controlled by a parameter temperature *t*, which determines the mobility of the system and is reduced by a positive factor $\alpha < 1$ in the algorithm. The likelihood of accepting a deteriorated solution decreases as the algorithm progresses. For a given value of *t*, some exchange trials D (repetitions) are performed until the value *t* is less than the final temperature δ. The initial temperature *t* should be initialized with *t* := $\varrho * \rho$, where ϱ is defined in Algorithm 1.

B. $PIRS^3A$: A Decentralized Scheme

Our objective is to design a decentralized scheme that helps each UE j to be associated with BS i that offers the highest throughput (i.e., high CQI, better channel conditions). We deploy $PIRS^3A$ in an Information Service Server (ISS) near MBS to solve the UA-RA problem in episodes, as illustrated in Algorithm 1. In each episode $e \in E$, $BS_i \in \{M_m \cup M_f\}$ informs ISS about its maximum capacity b_i, which represents the Knapsack capacity. Note that BSs which participate in episode e will not participate in the following episodes. Each UE $j \in N$ will inform ISS about the TBS index (obtained

Algorithm 1: $PIRS^3A$ for UA-RA Problem

```
foreach episode e ∈ E do
    Each BS i, i ∈ M informs ISS b_i
    foreach j ∈ N do
        Inform ISS about TBS Index q_ij
        if q_ij satisfies (4) then
            q_ij ∈ Q
        else
            UE j will not participate in the problem
        Inform ISS about p_j and r_ij
    initialization: P := [p_1, p_2, ..., p_ñ] where ñ ≠ n,
      ñ = length(Q); R:= [r_i1, r_i2, ..., r_iñ] where
      ñ ≠ n, , U_p:= {U_1, U_2, ..., U_ñ}, ∀j̃ ∈ U_p; X: = 0.
    Perform performance-improvement (Section III.E)
    Set values of parameters suggested in TABLE. IV
      i.e. an initial temp.(t) =
      (max{p_j̃|∀j̃} − min{p_j̃|∀j̃} * ρ); α; δ; D
    Select Initial Solution ω = (x_1, x_2, . . . , x_ñ) ∈ Υ;
    Incumbent Solution ← f(ω);
    repeat
        Set repetition counter a = 0;
        repeat
            Select integer ĩ from set {1, 2, ..., ñ};
            if x_ĩ = 0, pick item ĩ then
                while solution ω' is unusable, do
                    Drop another item from ω'
                      randomly; denote new solution as
                      ω';
                Let Δ = f(ω') - f(ω);
                while Δ ≥ 0 or Random (0,1) <
                  exp^(Δ/t_a), do
                    ω ← ω';
            else
                Drop item ĩ, and pick up another item
                  randomly, get new solution ω';
                Let Δ = f(ω') - f(ω);
                while Δ ≥ 0 or Random (0,1) <
                  exp^(Δ/t_a), do
                    ω ← ω';
            if f(ω') > incumbent solution then
                Incumbent solution ← f(ω');
            a = a+1;
        until a = D;
        Set t = α * t;
    until t > δ;
    N = N - ω';
```

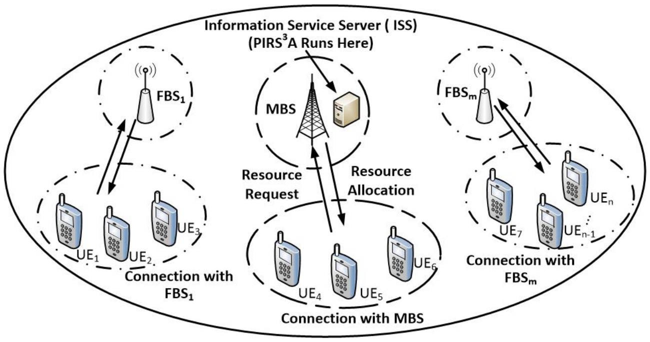

Fig. 2. System model of a two tier macro-femto HetNet for $PIRS^3A$.

by mapping) and the estimated available throughput p_j and demands r_{ij}. Let $U_p = Q \cup \tilde{Q}$ denotes the set of participating UEs where Q is the set of UEs meeting the constraint in Eq. (4) and $\tilde{Q}$ is the remaining UEs. Let X:= [x_1, x_2, ..., $x_{\tilde{n}}$] denotes a matrix of variable $x^i_{\tilde{j}}$ where $\tilde{j} \in U_p$. Note that only UEs in Q will be selected in the current episode. Vector P, denoting the estimated available throughput, vector R specifying the users demands (*e.g.,* throughput) and the set Q will then be created.

Each episode e includes a *diversification (exploration)* and an *intensification (exploitation)* phase. In the former, only UEs in Q are selected; The ones in $\tilde{Q}$ are considered in the next episode $e + 1$. In the latter, after solving Eq. (1) subject to Eqs. (2) and (3), UEs in Q will get associated with BS i, and resources will be allocated to them. After each episode, the set N is updated so that it contains only UEs in $\tilde{Q}$ and those who were not part of the obtained near-optimal solution $\omega^{'}$.

C. Available Throughput Estimation Method

First, BS i sends a control signal (CS) to each UE j that wants to be associated with it. Based on received CS, each UE j calculates CQI. Using CQI, MCS, and TBS mapping indices table [18], the j^{th} UE calculates TBS index along with its allocated RBs and the estimated available throughput that BS i can provide. The UE then sends this information to the ISS where our proposed scheme runs. If the received TBS index information satisfies Eq. (4), the UE will participate in the MKP problem. If selected after solving MKP, the UE will get associated with BS i and allocated the necessary resources.

D. Parameters Range Search Space Reduction

To perform algorithm fine-tuning, we select a set of parameters (*i.e.,* ρ, α, δ and D described in Section III.A) that intuitively appear to affect the efficiency of solving MKP solving using meta-heuristics SA. To generalize our results and compare them to one another, we use the relative deviation from the optimum expressed as follows:

$$\kappa(e) = \frac{|f(e^*) - f(e)|}{f(e^*)} \tag{6}$$

where *f(e)* is the computed solution while $f(e^*)$ is the best-known solution to the problem. Thus, the lower is the value

TABLE II
PARAMETER SETTINGS FOR SA.

SA param.	ρ	α	δ	D
Low	0.1	0.01	0.00001	0.1n
High	0.9	0.99	0.001	2n

TABLE III
RSM RESULTS FOR SA.

SA param.	Instance 1	Instance 2	Instance 3	Instance 4
ρ	0.628	0.7	0.7	0.7
α	0.745	0.255	0.745	0.255
δ	0.00041	0.00051	0.00051	0.00051
D	61	42	61	61

TABLE IV
DEFAULT AND SUGGESTED PARAMETER SETTINGS FOR SA.

SA param.	ρ	α	δ	D
Default	0.5	0.6	0.001	60
Suggested	0.8	0.71	0.000595	40

of $\kappa(e)$ for the meta-heuristic, the better is the performance of the algorithm [10]. The parameters and their necessary corresponding levels (low and high) required by a 2^g complete factorial are indicated in TABLE II. The fine-tuning of SA on MKP uses four arbitrary instances indicated in TABLE II and available in GitHub[2]. The $2^{g=4}$ [9] full factorial design was used to identify the factors that influence the algorithm the most. Thus, in the first step of our approach, we applied DoE ANOVA analysis [10] on the four parameters. From the analysis, we concluded that (i) ρ and α are significant for the algorithm regardless of the instance selected. δ and D are significant only in some instances; and (ii) the influence of every parameter varies depending on the instance studied. The next stage consists of applying RSM (Reduced Surface Methodology) to explore the neighborhood regions around a promising solution region.

In this context, RSM employs a sampling technique that finds the best match for each studied parameter to obtain a sub-optimal value that corresponds to estimated throughput in Eq. (1) with low convergence time. We define a range of values between each parameter's minimum and maximum from the RSM results presented in TABLE. III. This forms a space for the search of candidate configurations. The employed procedure consists of the simultaneous variation of all four parameters until ANOVA shows statistical significance. RSM results suggest an empirical model for the four parameters.

In the last step, we used the IRACE [21] package for racing algorithm implementation to select a good possible configuration out of many options. For this study, the settings used for SA are $\rho \in \{0.62, 0.7\}$, $\alpha \in \{0.255, 0.5, 0.745\}$, $\delta \in \{0.00041, 0.00051\}$, $D \in \{39, 52, 65\}$ as shown in TABLE. III. Because of the large difference between the minimum and maximum values obtained from the results of RSM, three different values, including the middle point for α and D are considered. Every possible combination leads to a different algorithm setting, so our search space consisted of 36 different

[2]https://github.com/bharat1992-bit

parameter settings for the SA algorithm. After applying the race algorithm in this search space, the best setting for the algorithm to solve MKP was found (see TABLE. IV[3]). DoE, RSM, and IRACE help obtain a reduced search space SA in the range of the parameters.

E. Solution Search Space Reduction

The solution search space also includes significantly large, useless, or infeasible solutions that the SA algorithm can consider when evaluating the target function. Undoubtedly, these annealing processes waste a lot of time in these useless solution regions and can easily impede the discovery of the correct solution. So, we remove the useless solution regions before applying SA to solve for MKP. To this end, we first classify n items according to the increasing profit p_j $(j = 1, ..., n)$. The greater the profit, the smaller the index, i.e., by descending order, these items were arrayed. If the number of items is greater than $\lfloor \frac{b}{n} \rfloor$, we only take the first $\lfloor \frac{b}{n} \rfloor$ items to participate in SA computation. The reason is that the items in $n - \lfloor \frac{b}{n} \rfloor$ are never selected when trying to obtain a maximum profit from a knapsack. So the remaining items $n - \lfloor \frac{b}{n} \rfloor$ belong to the useless solutions region. Hence, these items should be removed from the search space. In this way, reducing the search space of the parameter ranges and the solution space can achieve near-optimal solution while decreasing uncertainty.

IV. Performance Evaluation

A. Search Space Reduction

Fig. 3 illustrates the arithmetic mean (AM) of the ten runs of meta-heuristics (DSA, $PIRS^3A$) on ten different instances of the MKP while Fig. 4 depicts the distribution of near-optimal value $(f(e))$ per instance. DSA corresponds to the default simulated annealing without any fine-tuning of search space ofthe parameter ranges and the solution space. We observe that $PIRS^3A$ is the closest to the optimal solution compared to DSA. Indeed, DSA [10] is a single solution-based algorithm with less exploration and high exploitation, because of which it can easily be stuck in local minima. Parameter search space reduction helps finding the best starting solution while solution search space reduction helps removing the useless regions, saving significant time in finding the global optima. By combining these two aspects, $PIRS^3A$ achieves the global optima with low convergence time.

TABLE. V shows the execution time (ms) for $PIRS^3A$ and DSA when varying the number of items. $PIRS^3A$ incurs an average execution time that is 60% shorter than that of DSA for 100 items and around 69% shorter for 2000 items. This confirms that our proposed scheme has reduced complexity while incurring low overhead time.

[3] *Default* corresponded to settings of SA used in [10] while *Suggested* corresponds to settings of SA obtained from fine-tuning.

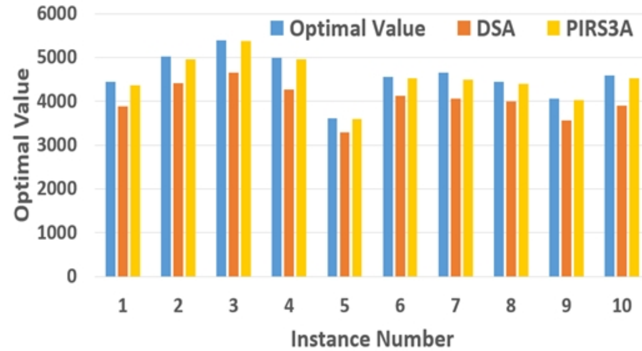

Fig. 3. Comparison of optimal values for two different versions of SA .

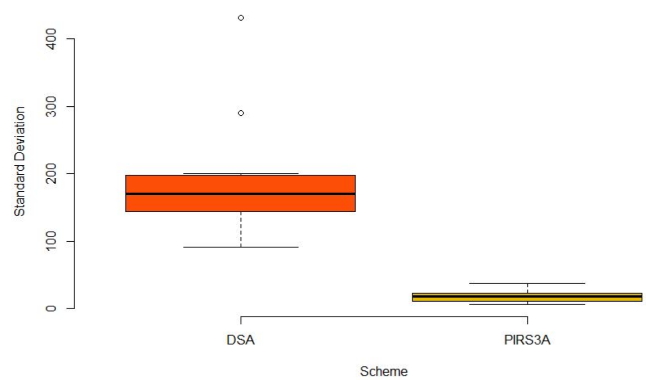

Fig. 4. Variability between two different versions of SA.

TABLE V
EXECUTION TIME OF DIFFERENT VERSIONS OF SA

Time(ms)	Number of Items					
Scheme	**50**	**100**	**200**	**500**	**1000**	**2000**
DSA	0.00028	0.00099	0.0035	0.025	0.090	0.36
$PIRS^3A$	0.00011	0.00040	0.0015	0.002	0.020	0.11

TABLE VI
DEFAULT SIMULATION PARAMETERS

Parameter	Value
Area of Region (G)	500m x 500m
UE traffic demand (ψ_j)	2 Mbps
number of BSs ($\|M\|$ = $\|M_m \cup M_f\|$)	11 = 1 + 10
Total transmit power of BSs	$\{46, 26\}$ dBm
Capacity of MBS	100.8 Mbps
Capacity of FBS	50.4 Mbps

B. QoS Assessment

We performed comprehensive simulations in NS-3 to evaluate our proposed algorithm. For all our experiments, we considered one MBS ($|M_m| = 1$) and ten FBSs ($|M_f| = 10$) that are deployed at fixed locations. All FBSs are initially switched off and are turned on sequentially when needed using CS. We randomly deployed UEs ($|N|$ = 70) following a homogeneous Poisson Point Process (PPP) for the different experiments and considered a discrete user demand (*i.e.,* requested data rate). To simulate channel fading, we used a log distance path loss model as in [19]. Other simulation parameters are depicted in TABLE VI. Fig. 5 depicts the QoS metrics perceived by users

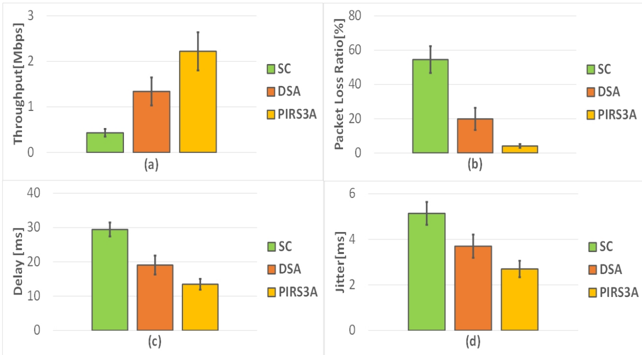

Fig. 5. QoS Metrics for 70 users under three different schemes.

in a HetNets environment. Beside $PIRS^3A$, we simulated two other schemes: Single-Cell (SC) and DSA. In SC, all UEs try to establish a connection with MBS only. DSA, on the other hand, corresponds to the classic SA scheme for the UA-RA problem formulated as MKP without fine-tuning in terms of parameter range and solution search space reduction.

Fig. 5(a) shows that $PIRS^3A$ incurred an average throughput of 2.22 Mbps which is 40% and 80% higher than DSA's (1.34 Mbps) and SC's (0.45 Mbps), respectively. Fig. 5(b) shows that $PIRS^3A$ incurred the least packet loss ratio (4%) compared to DSA (19%) and SC (54.5%). Fig. 5(c) illustrates that $PIRS^3A$ experienced the shortest delay (13.5 ms) which is 29.32% and 54% lower than DSA's (19 ms) and SC's (29.5 ms). Finally, Fig. 5(d) shows that $PIRS^3A$ produced the lowest jitter (2.7 ms) compared to DSA (3.7 ms) and SC (5.1 ms). The reason $PIRS^3A$ outperformed the remaining schemes in all the QoS metrics is its self-organized intrinsic property. Indeed, the parameter search space reduction helps selecting the best starting optimal solution as meta-heuristic algorithm for solving combinatorial optimization problems that heavily depend on the selected starting solution. In addition, the solution search space reduction helps increase the likelihood of selecting UEs with the highest estimated throughput, i.e., a high TBS index with a specific BS while considering other parameters such as BS capacity, UEs requirements and channel efficiency.

V. Conclusions and Future Work

This paper proposed $PIRS^3A$, an algorithm for solving the user association and resource allocation (UA-RA) problem in HetNets. The UA-RA problem was formulated as an MKP where BSs represent the knapsacks in this solution and UEs are the items to be fitted into the knapsacks. Simulation results show that the proposed solution outperforms alternative solutions in complexity, overhead time, and QoS metrics. Future work will integrate the dynamics of interference mitigation with UA and RA in a joint solution.

VI. Acknowledgement

This work received financial support from the Science Foundation Ireland (SFI) via grants 18/CRT/6224 (SFI Centre for Research Training in Digitally-Enhanced Reality D-REAL), 16/SP/3804 (Enable) and 12/RC/2289_P2 (Insight).

References

[1] Ericsson, "Ericsson Mobility Report November 2020," Ericsson, Nov. 2020. [Online]. Available: https://www.ericsson.com/en/mobility-report/reports/november-2020

[2] M. A. Togou et al., "DBNS: A Distributed Blockchain-Enabled Network Slicing Framework for 5G Networks," in *IEEE Communications Magazine*, vol. 58, no. 11, pp. 90-96, November 2020.

[3] B. Agarwal, M. A. Togou, M. Ruffini and G. -M. Muntean, "Mitigating the Impact of Cross-Tier Interference on Quality in Heterogeneous Cellular Networks," *2020 IEEE 45th Conference on Local Computer Networks (LCN)*, Sydney, NSW, Australia, 2020.

[4] A. Alnoman, L. Ferdouse and A. Anpalagan, "Fuzzy-Based Joint User Association and Resource Allocation in HetNets," *2017 IEEE 86th Vehicular Technology Conference (VTC-Fall),* Toronto, ON, 2017, pp. 1-5.

[5] F. Wang, W. Chen, H. Tang and Q. Wu, "Joint Optimization of User Association, Subchannel Allocation, and Power Allocation in Multi-Cell Multi-Association OFDMA Heterogeneous Networks," in *IEEE Transactions on Communications,* vol. 65, no. 6, pp. 2672-2684, June 2017.

[6] M. Feng, S. Mao and T. Jiang, "Joint Frame Design, Resource Allocation and User Association for Massive MIMO Heterogeneous Networks With Wireless Backhaul," in *IEEE Transactions on Wireless Communications,* vol. 17, no. 3, pp. 1937-1950, March 2018.

[7] L. Zhong, M. Li, Y. Cao and T. Jiang, "Stable User Association and Resource Allocation Based on Stackelberg Game in Backhaul-Constrained HetNets," in *IEEE Transactions on Vehicular Technology,* vol. 68, no. 10, pp. 10239-10251, Oct. 2019

[8] N. Sapountzis, T. Spyropoulos, N. Nikaein and U. Salim, "Optimal downlink and uplink user association in backhaul-limited HetNets," *IEEE INFOCOM 2016 - IEEE International Conference on Computer Communications,* San Francisco, USA, 2016, pp. 1-9.

[9] X. Luo, "Delay-Oriented QoS-Aware User Association and Resource Allocation in Heterogeneous Cellular Networks," in *IEEE Transactions on Wireless Communications,* vol. 16, no. 3, pp. 1809-1822, March 2017.

[10] E. Barbosa, E. Senne and M. Silva, "Improving the Performance of Metaheuristics: An Approach Combining Response Surface Methodology and Racing Algorithms," *International Journal of Engineering Mathematics,* vol. 2015, pp. 1-9, 2015.

[11] I. Akyar, *Wide spectra of quality control.* Rijeka, Croatia: InTech, 2011.

[12] A. I. Khuri and S. Mukhopadhyay, "Response surface methodology," *Wiley Interdisciplinary Reviews: Computational Statistics*, vol. 2, no. 2, pp. 128–149, 2010.

[13] M. Birattari , T. Stützle, L. Paquete, and K. Varrentrapp, "A Racing Algorithm for Configuring Metaheuristics," *Proceedings of the 4th Annual Conference on Genetic and Evolutionary Computation,* pp. 11–18, Jul. 2002.

[14] M. Birattari, Z. Yuan, P. Balaprakash, and T. Stützle, "F-Race and Iterated F-Race: An Overview," *Experimental Methods for the Analysis of Optimization Algorithms,* pp. 311–336, 2010.

[15] K. Satoda, E. Takahashi, T. Onishi, T. Suzuki, D. Ohta, K. Kobayashi, and T. Murase, "Passive Method for Estimating Available Throughput for Autonomous Off-Peak Data Transfer," *Wireless Communications and Mobile Computing,* vol. 2020, pp. 1–12, 2020.

[16] E. Yaacoub and Z. Dawy, "LTE radio network planning with HetNets: BS placement optimization using simulated annealing," *IEEE Mediterranean Electrotechnical Conf. (MELECON),* Beirut, Lebanon 2014

[17] Z. I. Kiss, A. C. Hosu, M. Varga and Z. A. Polgar, "Load balancing solution for heterogeneous wireless networks based on the knapsack problem," *International Conference on Telecommunications and Signal Processing (TSP),* Prague, 2015, pp. 1-6.

[18] 3GPP TR 21.916 Rel.16, Evolved Universal Terrestrial Radio Access (E-UTRA); Physical Layer Procedures, 2020.

[19] T. Z. Oo, N. H. Tran, W. Saad, D. Niyato, Z. Han and C. S. Hong, "Offloading in HetNet: A Coordination of Interference Mitigation, User Association, and Resource Allocation," in *IEEE Transactions on Mobile Computing,* vol. 16, no. 8, pp. 2276-2291, 1 Aug. 2017.

[20] A. I. Khuri, "A general overview of response surface methodology," *Biometrics & Biostatistics International Journal,* vol. 5, no. 3, pp. 87–93, 2017.

[21] M. López-Ibáñez et al., "The irace package: Iterated racing for automatic algorithm configuration," *Operations Research Perspectives,* vol. 3, pp. 43-58, 2016.