Performance Evaluation of the Angle Modulated Particle Swarm Optimization Algorithm in a Heterogeneous Network in Shared Spectrum Access

Anabel Martínez-Vargas[1], Ángel G. Andrade[2]

[1] Universidad Politécnica de Pachuca,
Mexico

[2] Universidad Autónoma de Baja California,
Facultad de Ingeniería,
Mexico

anabel.martinez@upp.edu.mx, aandrade@uabc.edu.mx

Abstract. Spectrum assignment (SA) controls the interference among secondary and primary users using spectrum sharing (SS) access. SA assigns the appropriate frequency band to a secondary user according to a predefined criterion. This work proposes a SA approach that controls the allocation channels to minimize network interference. The angle modulated particle swarm optimization (AMPSO) algorithm is applied to maximize the heterogeneous network (HetNet) throughput when secondary users exploit a channel simultaneously with a primary user. The AMPSO results are compared with the memory binary particle swarm optimization (MBPSO), the socio-cognitive particle swarm optimization (SCPSO), and the modified version of binary particle swarm optimization (ModBPSO). Comparison results showed that AMPSO is suited for scenarios with high quality of service (QoS) requirements and many secondary and primary users deployed in an area. AMPSO presents the best performance by maximizing spectrum reuse. It selects the secondary and primary users to share a communication channel and maximizes the total throughput in the HetNet.

Keywords. Angle modulated particle swarm optimization, spectrum assignment, spectrum sharing, heterogeneous network.

1 Introduction

The continued development of radio technology and new services has increased the world's dependence on wireless communications, growing the demand and the cost of the radio spectrum finite resource [31]. By 2023, over 70 percent of the global population will have mobile connectivity [6]. To address this growing challenge, regulators will require policies, new approaches, and technological innovations to enable flexible and efficient access to the radio spectrum.

Today, the static spectrum allocation policy regulates wireless networks. Regulators decide on the usage of a spectrum band, providing a license to each user to transmit on a frequency over a specific area. This rigid spectrum regulation guarantees that destructive interference among wireless technologies does not occur [4]. However, it has led to the under-utilization of the radio spectrum, as studies have pointed out [11]. In this context, SS becomes a promising approach to improving spectrum usage efficiency [3].

SS enables mobile users to use a frequency band in a specific geographical area from different wireless communication technologies. An SS network has two kinds of users: the primary users (PUs) and the secondary users (SUs). PUs have guaranteed access since they are licensed users. Consequently, SUs access the licensed spectrum if they do not harm the operation of the PUs. Hence, the PUs do not experience service degradation due to interference caused by the SUs [24].

An important issue of SS strategy is the QoS requirements concerning the signal-to-interference-noise ratio (SINR) for both PUs and SUs during the concurrent spectrum access. By facing this issue, no network tiers undergo service degradation due to the interference, achieving a peaceful coexistence.

SA is a key task to accomplish the SS approach. SA limits the interference between SUs and PUs operating in the same geographical area by assigning the appropriate frequency band to an SU by one or more criteria: interference/power, throughput, fairness, delay, price, energy efficiency, risk, and network connectivity [32].

After that, a suitable technique is selected to solve the objective(s) such as heuristics, graph theory, linear programming, fuzzy logic [22], evolutionary algorithms [25], swarm algorithms [33], etc.

For conventional cellular networks, the SS approach enlarges the pool of available spectrum resources for mobile users through femtocells (small cells), overlaid on the existing macrocell. That mixture of different types of cells is known as a HetNet [1]. However, in a HetNet deployment, reusing radio resources leads to destructive interference for macro-users (PUs) and femto-users (SUs) [14]. The unplanned positions of femto-base stations lead to two kinds of interference: cross-tier (the aggressor and the victim of interference belong to different tiers) and intra-tier (the aggressor and the victim of interference belong to the same tier) [5].

This work considers the underlay SS paradigm in a HetNet to propose a solution to the SA problem. Then, we maximize network throughput when one or more SUs reuse a channel simultaneously with the PU, satisfying QoS requirements. The SA problem belongs to the class of the NP-Hard problems, i.e., no known algorithm generates a guaranteed optimal solution in an execution time expressed as a finite polynomial of the problem dimension [32]. Therefore metaheuristics are suitable to tackle the SA problem by discarding solutions in polynomial time [30]. This work determines the maximum HetNet throughput from identifying SUs and PUs that have access to the same spectral band. That solution also ensures a peaceful coexistence among PUs and SUs in terms of interference. We apply metaheuristics to solve the SA problem in HetNet.

In this case, the binary optimization algorithm represents each solution as a binary string. The number of vector elements equals the number of SUs in the HetNet. If n SUs are deployed in HetNet, then the vector solution size is n bits. Therefore, the size of the binary search space doubles with each element (SU) added to the binary string (solution). It is envisioned that deploy ultra-dense small cells in the coverage region of macrocells will be a solution to the exponentially increased traffic in the following years [1].

In light of this, we deal with a high-dimensionality problem that enlarges the search space, increasing the computational complexity. The motivation for applying AMPSO to solve the SA problem is its ability to handle higher-dimensional problems [23]. AMPSO reduces a particle to a four-dimensional particle defined in continuous space, with a direct mapping back to binary space.

In previous work, we reported an admission control and channel allocation algorithm [19], based on the underlay shared mode and the MBPSO algorithm. However, from the results obtained in [19], we observed that when the number of SUs in the network increased (high dimensionality), the MBPSO algorithm used did not converge to a good solution because the optimization complexity of the SA problem increased. The AMPSO algorithm offers a way to reduce the complexity of binary problems faster than conventional BPSO algorithms. Therefore, we consider applying the AMPSO technique to solve the SA problem in scenarios with a high density of SUs and QoS requirements in the wireless network. The purpose of our work is to evaluate the efficacy of the AMPSO algorithm to find a solution in those complex scenarios.

Other studies have addressed the throughput maximization in HetNets. For example, work in [29] considers an LTE HetNet composed of femtocells and macrocells. It proposes a centralized scheduling approach to mitigate interference and maximize the throughput of the HetNet. Then an optimization problem is formulated as a mixed-integer non-linear programming problem

(MINLP). Given that the MINLP is NP-Hard, it is transformed to be solved in polynomial time using a heuristic algorithm inspired by sociological theory. This transformation only applies to a scenario that authors call an apartment environment (OAE) with obstructive structures.

Work in [26] addresses resource allocation in a HetNet composed of one macrocell and several femtocells. It aims to maximize the femto-tier throughput. To reduce the complexity, the authors divide the maximization problem into two sub-problems: the clustering problem and the resource allocation problem. The first problem that forms the femtocell groups is solved by using an evolutionary game. In contrast, the second problem is posed as one of maximization of the throughput within a cluster. By doing this, it is possible to address it by the particle swarm optimization (PSO) technique.

In contrast, work in [27] addresses the SS with the primary objective of increasing the sum throughput system using QoS constraints for both SUs and PUs. They solve the SA problem by applying particle swarm optimization (PSO) in a homogeneous network (802.11), i.e., cells with the same characteristics. Then an optimal relay selection method is coupled. However, work in [15] envisions that the corresponding number of base stations in the network will increase as the number of users increases. So, it emphasizes that the design of the SS techniques must keep in view picocells, femtocells, small cells, etc., simultaneously in the network.

In [36], the authors maximize the D2D users' throughput with minimal interference to the cellular users. This is done in a multi-tier HetNet. Then, the authors propose an autonomous spectrum allocation scheme with distributed Q-learning. The D2D users can learn the wireless environment and select spectrum resources autonomously to achieve the objective through this strategy. The D2D users operate the underlay shared mode, i.e., they reuse spectrum used by cellular users. The authors simulated their scheme using the Monte Carlo technique by executing 10000 runs.

Finally, the study [17] proposes a numerical approach of coexisting LTE and WiFi networks to share an unlicensed spectrum. It maximizes total throughput in a HetNet if an access point (AP) achieves a throughput threshold level. Then, it applies decentralized and centralized traffic management schemes to show a maximum per-user link throughput of an AP and per-user network throughput.

The authors characterize the statistical property of the cell load and channel access probability of each AP in a low-complexity form. The per-user link throughput and per-user network throughput are based on the derived mean spectrum efficiencies and maximize them applying Shannon transform to a non-negative random variable. The simulation results conclude in both schemes that offloading traffic from the LTE network to the WiFi network initially improves the per-user network throughput, but it finally leads to its reduction due to too much offloading.

Unlike works [26] and [29], we do not apply any transformation to the objective function to convert the problem into a deterministic problem. Through AMPSO, we handle candidate solutions with high dimensionality. As works in [36] and [27], we also consider QoS constraints in SUs and PUs, i.e., we guarantee successful communication to both kinds of users. Just as works [29] and [17], we assume centralized management in which our proposed approach is processed in the macro base station.

This paper has the following structure: Section 2 presents the system model and the problem statement. Section 3 describes AMPSO. Section 4 describes AMPSO to resolve the SA problem. Section 5 shows simulation results and consequently, Section 6 presents a discussion. Finally, Section 7 concludes the paper and addresses the implications for further research.

2 System Model and Problem Formulation

Fig. 1 is the down-link scenario considered in this work. It is a HetNet where femto-cells (the red dashed circles) exist within the coverage area A of a macrocell (the black dashed circle). The macrocell has a macro-base station (MBS) which communicates with its associated macro-users (PUs). Consequently, the union of a transmitter, i.e. an MBS, and a receiver (a macro-user) is referred to as a primary link. In Fig. 1, the primary links are the black arrows; each primary link is identified by a number beside the link (the green numbers). Also, each primary link has a primary channel assigned (the number in brackets). The total number of primary links in A is Pl. The primary links have fixed locations. On the other hand, the femto-cell has a femto-base station (FBS) which communicates with its attached femto-user (SU). Then, a secondary link consists of the union of a transmitter (i.e. an FBS) and its corresponding receiver (a femto-user). Each secondary link is identified by a number beside the link (the blue numbers in Fig. 1). Then a primary channel is assigned to several secondary links. The primary channels assigned to each secondary link are the number in braces in Fig. 1. The total number of secondary links in A is Sl.

We assume that FBSs do not have channels to assign to their femto-users, so macro-users must share their primary channels. In the beginning, primary channels are assigned randomly to secondary links. In Fig. 1, we show the case when primary channel 1 is shared among secondary and primary links. The red number 1 means that primary channel 1 is being shared among secondary links 3, 4, 5, and primary link 2.

This channel assignment will generate a level of interference between these links, and network capacity will be affected. In the worst case, if the interference exceeds a predefined QoS threshold, this channel assignment will not be valid. Then, it will be necessary to find another configuration to assign channel 1. Also in Fig. 1, other primary channels are being shared. For example, primary channel 4 is being shared between secondary link 1 and primary link 1. Another example is primary channel 3 that is being shared between secondary link 2 and primary link 4.

The SINR (in dB) is the instantaneous ratio of desired energy to interference. It is a metric on a receiver. In single-hop communications, the SINR must accomplish a minimum SINR threshold to indicate a successful reception [2]. Then, SINR relates to QoS. If primary links experience dropped calls or cannot connect because of the high interference due to the presence of the secondary links in the geographical region, the aim of SS is not achieved at all. Consequently, each service has a QoS or SINR threshold to achieve. For example, a voice service has a target QoS of 3 dB to be considered a successful communication between the transmitter and the receiver.

The SINR in a macro-user of a primary link v is given by [20]:

$$SINR_v = (P_v/ldp(v)^n)/(\sum_{k\in\varphi} P_k/dps(k,v)^n), \quad 1 \leq v \leq Pl. \quad (1)$$

where P_v is the transmit power of the primary link v. $ldp(v)$ is the link distance of the primary link v. n is the path loss exponent (a value between 2 and 4). Those parameters characterizes the desired signal. Consequently, φ is the set of the interferers, i.e. the active secondary links that have assigned the same primary channel as the primary link v. k is the index of interferers. P_k is the transmit power of the secondary link k. $dps(k,v)$ is the distance from the transmitter in secondary link k to the receiver in primary link v.

In Fig. 1, $SINR_v$ is computed in the macro-user of the primary link 2. There, the macro-user in primary link 2 has three interferers: secondary links 3, 4, and 5. The aforementioned is the aggregated cross-tier interference, i.e., the total interference from the secondary links that attempt to simultaneously exploit a channel with the primary link v.

Similarly, the SINR in a femto-user of a secondary link u is given by [20]:

$$SINR_u = (P_u/lds(u)^n)/(\sum_{k\in\varphi} P_k/dss(k,u)^n + P_v/dps(v,u)^n), 1 \leq u \leq Sl. \quad (2)$$

where P_u is the transmit power of the secondary link u. $lds(u)$ is the link distance of the secondary link u. n is the path loss exponent (a value between 2 and 4). Those variables characterize the desired signal. Meanwhile, φ is the set of the interferers, i.e., the active secondary links that have assigned the same primary channel as the secondary link u. k is the index of interferers. P_k is the transmit power of the transmitter of secondary link k. $dss(k,u)$ is the distance from the transmitter

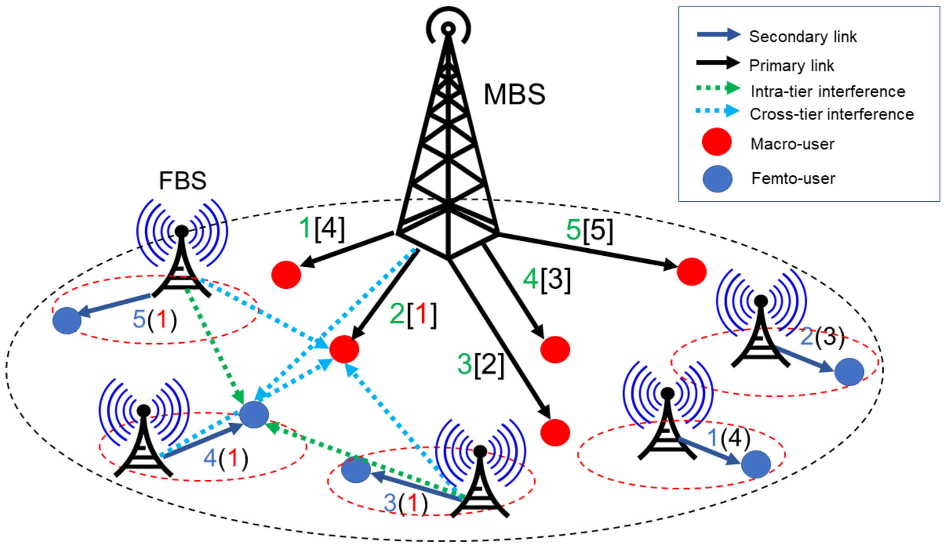


Fig. 1. HetNet scenario

of secondary link k to the receiver of secondary link u. Therefore, those parameters represent the aggregate intra-tier interference.

For example, in Fig. 1, the intra-tier interference on the femto-user of the secondary link 4 comes from the secondary links 3 and 5. Likewise, P_v is the transmit power of the interferer primary link v (it has assigned the same primary channel as secondary link u). $dps(v,u)$ is the distance from the transmitter of the primary link v to the receiver of the secondary link u. Those parameters characterize the cross-tier interference perceived by a receiver of secondary link u. In Fig. 1, the cross-tier interference on the femto-user of the secondary link 4 comes from the primary link 2. In Fig. 1, $SINR_u$ is computed in secondary link 4.

Positive values of SINR indicate that the desired signal is greater than the interference. On the other hand, negative values of SINR refer to that the interference is greater than the desired signal.

Data rate (in Mbps) of the secondary link u and the primary link v are described in equations (3) and (4), respectively [20]:

$$C_u^{'} = Blog_2(1 + SINR_u), \tag{3}$$

$$C_v^{''} = Blog_2(1 + SINR_v), \tag{4}$$

where B is the channel bandwidth that secondary and primary links share. Positive values of SINR result in better throughput. In contrast, negative values of SINR lead to worse throughput.

We aim to optimize the sum throughput in the SS network. We formulate the optimization problem as [20]:

$$\text{Maximize} \sum_{u=1}^{Sl} c_u^{'} \cdot x_u + \sum_{v=1}^{Pl} c_v^{''}, \tag{5}$$

Subject to:

$$SINR_u \geq \gamma, \tag{6}$$

$$SINR_v \geq \alpha, \tag{7}$$

$$c_u^{'} > 0, u = 1, 2, ..., Sl, \tag{8}$$

$$c_v^{''} > 0, v = 1, 2, ..., Pl, \tag{9}$$

$$c_u^{'}, c_v^{''}, \in \mathrm{R}^+, \tag{10}$$

$$x_u = \begin{cases} 1, if\ SINR_u \geq \gamma\ and\ SINR_v \geq \alpha \\ 0, otherwise. \end{cases} \tag{11}$$

The task is to find a binary vector x_u=$(x_1, ..., x_{Sl})$ for which the objective function in equation (5) is maximum. Equation (5) represents the sum throughput of the SS network. It takes into account the selected secondary links, x_u, along with the primary links that coexist in the same region and share the same spectrum. Equations (6) and (7) are the SINR requirements of the secondary links and primary links respectively.

A successful transmission in the primary link v is achieved if it reaches the SINR threshold α. Similarly, a successful transmission in the secondary link u is reached if its SINR is above the SINR threshold γ. Each position u in the binary vector x in equation (11) symbolizes if secondary link u is related to the primary link v ($x_u = 1$) or not ($x_u = 0$).

3 Angle Modulated Particle Swarm Optimization Algorithm

AMPSO [23] is an alternative version of binary particle swarm optimization (BPSO) [13] to address high dimensionality problems.

To do so, AMPSO employs standard PSO to optimize the coefficients of the following trigonometric function:

$$g(x) = sin[2\pi(x-a)b \cdot cos(2\pi(x-a)c)] + d]. \quad (12)$$

The function in (12) is called the generating function, and it is used as a bit string generator. To optimize the coefficients of the generating function, the position of a particle i is composed of a four-dimensional vector $X_i = (a_i, b_i, c_i, d_i)$. The coefficient a controls the horizontal shift of the entire function. The coefficient b influences the maximum frequency of the sine wave and controls the amplitude of the cosine wave. The coefficient c affects the frequency of the cosine wave (which changes the rate at which the frequency of the sine function changes), and d controls the vertical shift of the function.

For example, Fig. 2 shows the evaluation of $g(x)$ in the [-2, 2] interval for a set of default coefficient parameters: a=0, b=1, c=1, d=0. The coefficient parameters are substituted in equation (12) to generate the bit string.

Then the function $g(x)$ is sampled n_b times, where n_b is the number of bits required to represent the solution. If the $g(x)$ value is positive, the corresponding bit is set to 1. Otherwise it is set to 0. A bit is generated for each interval evaluated, so that each set of coefficient parameters that composes X_i has a Xb_i bit string with it.

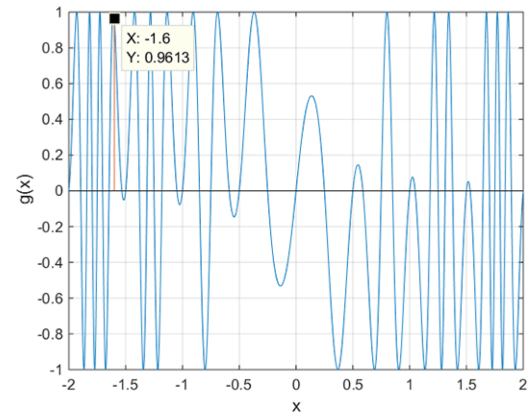


Fig. 2. Angle modulation function $g(x)$ for default parameters a=0, b=1, c=1, and d=0

For example, to generate 10 bits from Fig. 2, we define 10 equal separated intervals: x_j= (-1.6, -1.2, -0.8, -0.4, 0, 0.4, 0.8, 1.2, 1.6, 2). Then we evaluate $g(x)$ for x_1=-1.6, and as Fig. 2 shows, the $g(x)$ value is positive, so the first bit of the particle is set to 1. This process is repeated for all the remaining values of x.

Once we have sampled all the values, the whole bit string is generated Xb_i=(1 0 0 1 0 0 1 1 0 0). Xb_i represents one of the possible solutions to the discrete problem, then, it is evaluated to assign a fitness value.

AMPSO updates the velocity v_{id} using equation (13) and position x_{id} using equation (14) of X_i according to conventional PSO [12]:

$$v_{id} = wv_{id} + c_1 r_{1d}(p_{id} - x_{id}) + c_2 r_{2d}(p_{gd} - x_{id}), \quad (13)$$

$$x_{id} = x_{id} + v_{id}, \quad (14)$$

where c_1 and c_2 are positive constants used to scale the contribution of the cognitive and social components. r_1 and r_2 are vectors of random values in the range [0, 1] which are sampled from a uniform distribution and per dimension.

AMPSO reduces a high dimensional bit string to a four-dimensional vector. Algorithm 1 describes AMPSO for maximizing goodness.

Algorithm 1: AMPSO

Data: The equal intervals x_j to sample generating function, the swarm size S, the initial values of the four coefficients a, b, c and d

Result: The best solution Pb_g, evaluation of the best Pb_g in the fitness function $f(Pb_g)$

Initialize position vector X_i, intervals vector x_j, velocity vector V_i, memory vector P_i = X_i;
repeat
    for *each particle i = 1 to number of particles in swarm S* **do**
        Function *Generate_bit_string_Xbi()*;
        Evaluate Xb_i using the objective function;
        Let $f(X_i)$=$f(Xb_i)$;
        Function *Generate_bit_string_Pbi()*;
        Evaluate Pb_i using the objective function;
        Let $f(P_i)$=$f(Pb_i)$;
        if *f(Xb_i) $>$ f(Pb_i)* **then**
            for *d = 1 to 4* **do**
                p_{id} = x_{id};
            end
        end
        g=i;
        for *each particle j = 1 to number of particles in swarm S* **do**
            Function *Generate_bit_string_Pbj()*;
            Evaluate Pb_j using the objective function;
            Let $f(P_j)$=$f(Pb_j)$;
            Function *Generate_bit_string_Pbg()*;
            Evaluate Pb_g using the objective function;
            Let $f(P_g)$=$f(Pb_g)$;
            if *f(Pb_j) $>$ f(Pb_g)* **then**
                g=j;
            end
        end
        for *d = 1 to 4* **do**
            Update velocity according to the equation (13);
            $v_{id} \in (-V_{max}, V_{max})$;
            Update position according to the equation (14);
        end
    end
until *stopping condition(s) satisfied*;

4 Angle Modulated Particle Swarm Optimization Algorithm to Resolve the Spectrum Assignment Problem

When a scenario (or snapshot) is analyzed using algorithm 3, Xb_i specifies a potential solution to solve the SA problem; that is, the set of secondary links that may coexist with the primary links in area A to achieve the maximum throughput, subject to QoS constraints for primary-secondary networks.

In algorithm 3, we include two new vectors: X_i' and P_i'. Each Xb_i has an X_i' which holds the candidate primary channels for the chosen secondary links. Take for example the snapshot illustrated in Fig. 3. Particle Xb_i indicates that secondary links 1, 3, 5, 6, 7, and 8 are

Algorithm 2: AMPSO, Functions

Function *Generate_bit_string_Xbi()*
  for *j=1 to* n_b **do**
    if *g(*X_i,x_j*)* ≥ 0 **then**
      xb_{ij}=1;
    else
      xb_{ij}=0;
    end
  end
End
Function *Generate_bit_string_Pbi()*
  for *j=1 to* n_b **do**
    if *g(*P_i,x_j*)* ≥ 0 **then**
      pb_{ij}=1;
    else
      pb_{ij}=0;
    end
  end
End
Function *Generate_bit_string_Pbj()*
  for *k=1 to* n_b **do**
    if *g(*P_j,x_k*)* ≥ 0 **then**
      pb_{jk}=1;
    else
      pb_{jk}=0;
    end
  end
End
Function *Generate_bit_string_Pbg()*
  for *k=1 to* n_b **do**
    if *g(*P_g,x_k*)* ≥ 0 **then**
      pb_{gk}=1;
    else
      pb_{gk}=0;
    end
  end
End

selected as a part of the solution; hence, X_i' is the channel allocation for those chosen secondary links. Consequently, P_i' keeps the best channels allocations find so far for Pb_i.

In contrast, the spectrum status vector holds the channel allocation for the primary links, so that, spectrum status vector is kept fixed through search. Mapping of X_i' and spectrum status

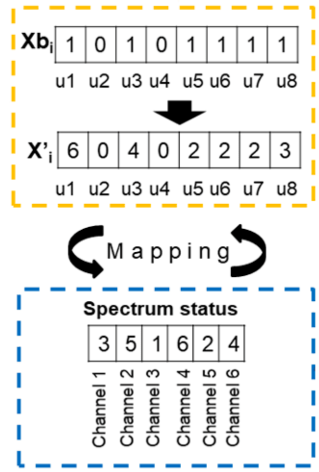


Fig. 3. Representation of particles for a given spectrum sharing access

provides the potential channels to share among the PUs and SUs. From the example in Fig. 3, channel 2 is exploited concurrently by primary link 5 and the secondary links 5, 6, and 7.

Once the bit strings Xb_i and Pb_i are generated as indicated in STEP 8 and STEP 9, the SINR levels for candidate SUs in Xb_i and PUs in the spectrum status vector are calculated. This is done by mapping each element of X_i' to its corresponding channel in the spectrum status vector. Also, the SINR levels for the best candidate SUs in Pb_i and PUs in the spectrum status vector are computed.

Those SINR levels are calculated by using equations (1) and (2). In STEP 12, if SINR restrictions in equations (6) - (7) are achieved for SUs and PUs, Xb_i and Pb_i are feasible solutions. Then their fitness values are calculated as shown in equation (5). Otherwise, if SINR levels are not achieved, Xb_i and Pb_i are infeasible solutions that are penalized by setting their fitness values to zero.

From STEP 14 to STEP 19, the process of finding the best set of secondary links so far by the i-th particle is performed. Consequently, the best position and the best channel allocation are kept in those steps.

Algorithm 3: AMPSO to resolve the spectrum assignment problem

Data: The total number of secondary links Sl, the total number of primary links Pl, the SINR thresholds γ = α, the swarm size S, the set of primary channels PC, the number of iterations T_{max}, the equal intervals x_j to sample generating function, the coefficients a, b, c and d

Result: The maximum data rate in the system $f(Pb_g)$, the set of selected secondary links Pb_g, the channel allocation for primary links in vector spectrum status, the best channel allocation for secondary links $P'g$

Locate randomly the total number of secondary links Sl and the total number of primary links Pl over the coverage area A;
Initialize randomly velocity vector V_i where $v_{id} \in (-V_{max}, V_{max})$. Set P_i = X_i;
Let coincide the personal best channel allocation vector P'_i and candidate channel allocation vector X'_i;
Initialize randomly vector spectrum status with values from 1 to Pl;
Initialize position vector X_i where $x_{id} \in$ (-1, 1);
repeat
    for *each particle i = 1 to number of particles in swarm S* **do**
        Function *Generate_bit_string_Xbi()*;
        Function *Generate_bit_string_Pbi()*;
        Compute SINR at SUs and PUs by mapping X'_i and spectrum status;
        Compute SINR at SUs and PUs by mapping P'_i and spectrum status;
        Evaluate Xb_i and Pb_i at the fitness function at (5) and restrictions from (6) to (11);
        Let $f(P_i)$=$f(Pb_i)$ and $f(X_i)$=$f(Xb_i)$;
        if *f*$(Xb_i) >$ *f*(Pb_i) **then**
            Perform from STEP 11 to STEP 13 from the algorithm (1);
            for *j=1 to* n_b **do**
                p'_{ij} = x'_{ij}
            end
        end
        g=i;
        for *each particle j = 1 to number of particles in swarm S* **do**
            Function *Generate_bit_string_Pbj()*;
            Function *Generate_bit_string_Pbg()*;
            Compute SINR at SUs and PUs by mapping P'_j and spectrum status;
            Compute SINR at SUs and PUs by mapping P'_g and spectrum status;
            Evaluate Pb_j and Pb_g at the fitness function at (5) and restrictions from (6) to (11);
            Let $f(P_g)$=$f(Pb_g)$ and $f(P_j)$=$f(Pb_j)$;
            if *f*$(Pb_j) >$ *f*(Pb_g) **then**
                g=j;
            end
        end
        Perform from STEP 27 to STEP 31 from the algorithm (1);
        for *k=1 to* n_b **do**
            Allocate randomly a new channel to x'_{ik} from the set of primary channels PC;
        end
    end
until *number of iterations* $< T_{max}$;

Another search process is performed from STEP 21 to STEP 31 to search for the best performer in the swarm.

The position and velocity of i-th particle are updated as shown in STEP 32.

The loop from STEP 33 to STEP 35, updates X_i'. The set of primary channels PC equals the channels in spectrum status vector. For example, from Fig. 3, they are five channels in spectrum status vector, so, $PC = \{1, 2, 3, 4, 5\}$.

Finally, in STEP 37, algorithm (3) repeats the above process until the maximum number of iterations T_{max} is met. Then, the solution of the problem is Pb_g which is the set of selected secondary links that maximize the throughput $f(Pb_g)$ with the primary links deployed in the area. Those selected secondary links also satisfy the QoS constraints, needed to keep the interference to a tolerable level for both secondary links and primary links.

5 Experimental Evaluation

In the following subsections, we present the scenario conditions to analyze AMPSO in the HetNet with SS approach. Then, we show the results obtained by the AMPSO for maximum throughput. For comparison, the SCPSO algorithm, the MBPSO algorithm [35], and the ModBPSO algorithm [34] are also included to solve the SA problem. Finally, we perform the Wilcoxon signed ranks and the sign test for multiple comparisons to confirm whether the AMPSO offers a significant improvement, or not, over the remaining BPSO variants.

5.1 Experimental Condition

We consider the downlink analysis of Fig. 1, characterized by a fixed deployment of primary links and a random deployment of secondary links in a 5000 m x 5000 m grid. An experiment is the combination of Pl, Sl, and SINR thresholds (γ = α). For each experiment, a BPSO variant is run for 500 independent instances (snapshots of random secondary links location and fixed primary links location). We incrementally vary the number of secondary links in the area, i.e., in step sizes of 10 secondary links. By doing that, interference rises gradually. The QoS requirement of 4 dB represents the less challenging scenario for the BPSO variants.

Therefore, most of the SUs deployed in the area will achieve that SINR threshold, i.e., most of them will be selected by the BPSO variants. On the other hand, the SINR threshold of 10 dB has a medium complexity for the BPSO variants.

That means that some SUs may be able to be above the SINR threshold of 10 dB. Finally, the SINR threshold of 14 dB is the most challenging scenario for the BPSO variants due to the high QoS requirement. As more primary and secondary links are in the coverage area, the interference can rise to a harmful level. Then it is more challenging for the BPSO variants to leverage it up to a tolerable level. At this point, the task of selecting secondary users is vital since it is the strategy that the BPSO variants apply to cope with interference.

In regards to the HetNet, the femto-user is set to a maximum radius of 30 m away (for minimizing attenuation due to loss path) from the FBS; whereas, the macro-user is deployed 1000 m away from the MBS. We assume that secondary links and primary links employ unit transmission power and homogeneous traffic. Multipath and shadow fading are not considered for the SINR calculation. The number of channels to share depends on the number of primary links deployed in the area.

Table 1 and Table 2 show the parameters used for the BPSO variants and the experiments respectively. Increasing the number of primary and secondary links in the HetNet under different QoS requirements (4, 10, and 14 dB) result in increasing the complexity to find a good solution for every BPSO variant.

For instance, cognitive factor c_1, social factor c_2, socio-cognitive factor c_3, inertia weights w^1 and w which are parameters for SCPSO were set as suggested in the study [7]. The lower bound w_{min} and the upper bound w_{max} which are parameters of MBPSO were set as proposed in [35]. Concerning ModBPSO, the mutation rate r_{mu} is set as suggested in [34]. Consequently, we set the parameter maximum velocity V_{max} as suggested in [13].

The simulation methodology is in Fig. 4. Once we set the parameters for a BPSO variant and experiment, the admission control and channel allocation algorithm based on a BPSO variant generates a snapshot of a HetNet scenario, and then the BPSO variant is run to solve equations (5) - (11). After the admission control and channel allocation algorithm based on a BPSO variant finishes its execution, it computes the maximum throughput for that snapshot. If the admission control and channel allocation algorithm based on a BPSO variant achieves the total sample snapshots to analyze, it selects the sample snapshot with the highest throughput.

Table 1. Parameters used for experiments

Parameters	Values
Number of secondary links Sl	10:100:10
Number of primary links Pl	6, 12, 24
Runs	500
SINR thresholds γ = α	4, 10, 14 dB
Channel bandwidth B	20 MHz

Table 2. Parameters used for the BPSO variants

Parameters	Values
Swarm size S	40
Maximum number of iterations T_{max}	100
Cognitive, social factors c_1, c_2	2, 2
Socio-cognitive factor c_3 (for SCPSO)	12
w_{max}, w_{min} (for MBPSO)	1.4, 0.1
$Iter_{max}$ (for MBPSO)	20
Inertia weight w^1 (for SCPSO)	0.9
Mutation rate r_{mu} (for ModBPSO)	0.02
Inertia weight w	0.721
Maximum velocity V_{max}	[-6, 6]

5.2 Experimental Results

In Figs. 5a, 5c, and 5e, we show the best solutions found by the BPSO variants for the HetNet when Pl = 6 at γ, α = [4, 10, 14] dB. The best solutions found by the AMPSO outperform the ones found by the remaining BPSO variants, in the range of 10-60 SUs. The ModBPSO comes next, however, for higher values of γ, α (high QoS requirements); it cannot find a solution. In Figs. 5b, 5d, and 5f, we show the average maximum data rate, i.e., we average the results over all samples in the experiment.

While the curves of ModBPSO, MBPSO and SCPSO come down in the range 30 - 40 SUs deployed in the area, AMPSO keeps almost a constant throughput with the highest data rates. When the other BPSO variants fail to find a solution, as in Fig. 5f which is the most challenging scenario, AMPSO is able not only in finding a solution but also in offering the one with the highest data rate.

Concerning the experiment when Pl = 12 at γ, α = [4, 10, 14] dB, the best solutions found and the average data rate are shown in Figs. 6a – 6f. In Figs. 6a, 6c, and 6e, AMPSO still outperforming the other BPSO variants. In Fig. 6e, while MBPSO, ModBPSO, and SCPSO could not find a solution in the range 80-100 SUs, AMPSO can find it. Figs. 6b, 6d, and 6f show that AMPSO produces, on average better solutions and it can find solutions even in the most challenging scenario as plotted in Fig. 6f.

Figs. 7a, 7c, and 7e are the best solutions found by the different versions of BPSO when Pl = 24 at γ, α = [4, 10, 14] dB. In this context, AMPSO performs better than the other BPSO variants, even when the QoS requirement is the highest, as in Fig. 7e. From that plot, we observe that AMPSO can find a solution when the others fail, especially in the range of 80 - 100 SUs. Figs. 7b, 7d, and 7f shows that AMPSO significantly outperforms the other BPSO methods in average throughput, finding solutions when the other BPSO methods cannot offer one as in Fig. 7f.

5.3 Use of Nonparametric Statistics for Comparing the Results

We perform the Wilcoxon signed ranks and the sign test for multiple comparisons to confirm whether AMPSO offers a significant improvement, or not, over the remaining BPSO variants for the HetNet with SS approach. Among the experiments, we are particularly interested in ones when $Sl = 100$. We summarize the results obtained for each experiment and BPSO variant in Table 3.

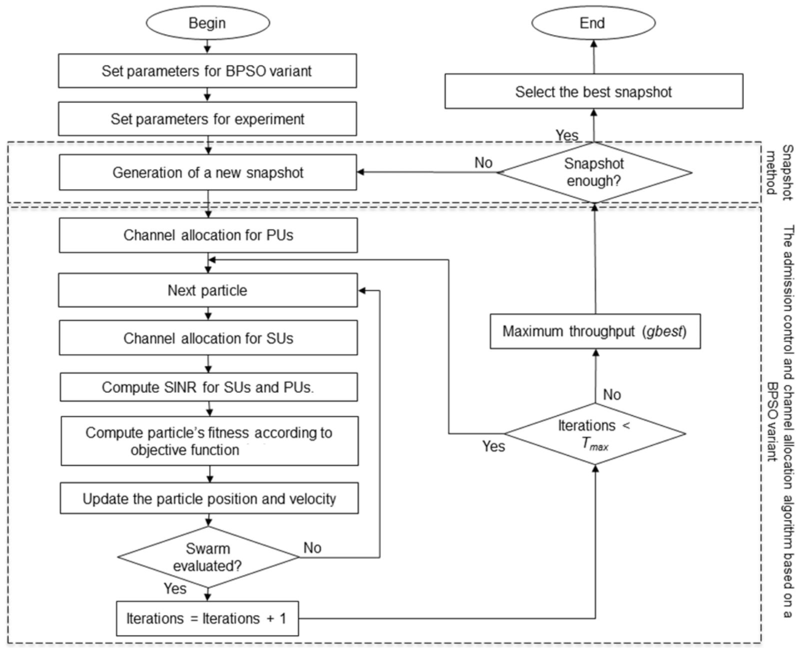


Fig. 4. Simulation methodology

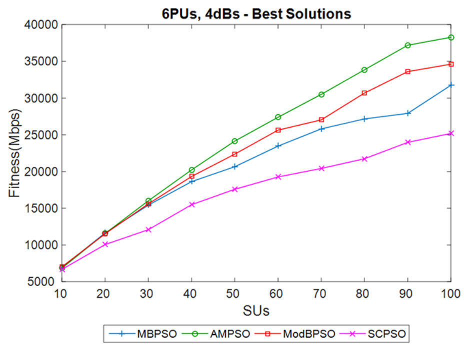
Table 3. Average fitness obtained in the different experiments

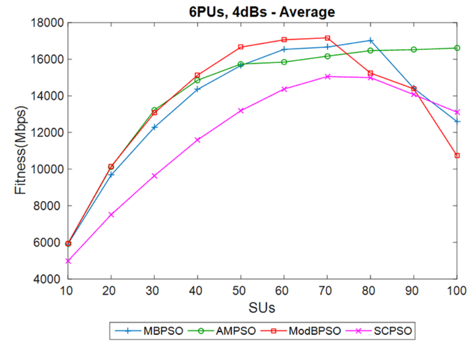
Experiment	AMPSO	MBPSO	ModBPSO	SCPSO
6 PUs, 100 SUs, 4 dB	16610.29	12599.9	10737.3	13109.61
6 PUs, 100 SUs, 10 dB	8913.5	440.2	617.41	832.44
6 PUs, 100 SUs, 14 dB	4992.05	80.9	0	114.26
12 PUs, 100 SUs, 4 dB	20519.99	19506.12	17721.69	18941.38
12 PUs, 100 SUs, 10 dB	10769.62	594.31	524.72	852.07
12 PUs, 100 SUs, 14 dB	5957.41	0	0	0
24 PUs, 100 SUs, 4 dB	24585.18	26138.38	24820.07	24787.52
24 PUs, 100 SUs, 10 dB	13151.15	947.02	1113.33	1409.48
24 PUs, 100 SUs, 14 dB	7302.51	0	0	0

The performance measure is the average fitness (throughput). Firstly, we present a comparative study on AMPSO performance and the remaining BPSO variants through pairwise comparisons. We apply the Wilcoxon signed ranks since it is a safe and robust nonparametric test for pairwise statistical comparisons. Also, the outliers (exceptionally good/bad performances) have less effect on it [8]. Table 4 summarizes the results of applying it, displaying the sum of rankings obtained in each comparison and the p-value associated.

Table 4. AMPSO shows improvement over SCPSO, ModBPSO, and MBPSO, with a level of significance α=0.05

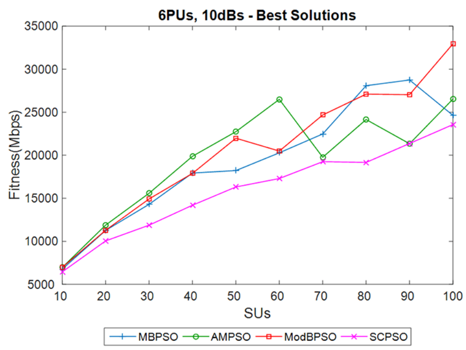
AMPSO vs.	R^+	R^-	p-value
SCPSO	44	1	0.011
ModBPSO	44	1	0.011
MBPSO	43	2	0.015

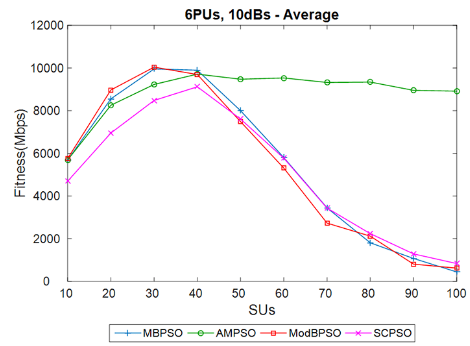




(a) best solutions γ, α = 4 dB

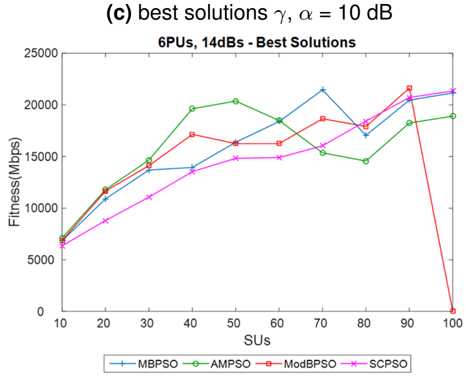
(b) average γ, α = 4 dB

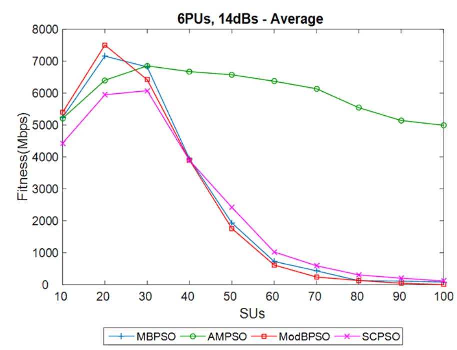




(c) best solutions γ, α = 10 dB

(d) average γ, α = 10 dB





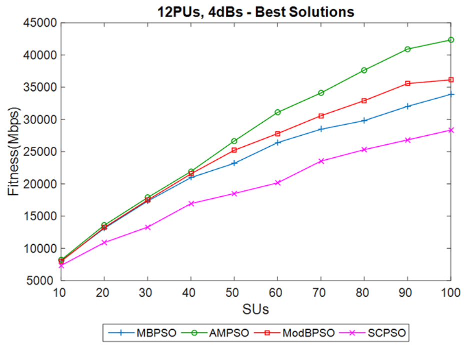
(e) best solutions γ, α = 14 dB

(f) average γ, α = 14 dB

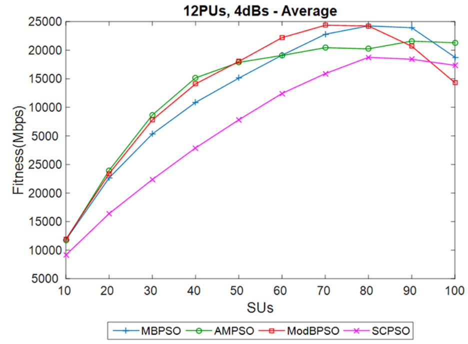
Fig. 5. System throughput of the BPSO variants when Pl = 6 at γ, α = [4, 10, 14] dB

As Table 4 states, AMPSO shows a significant improvement over SCPSO, ModBPSO, and MBPSO, with a level of significance α=0.05. Since p-values are less than α=0.05, we reject the null hypothesis.

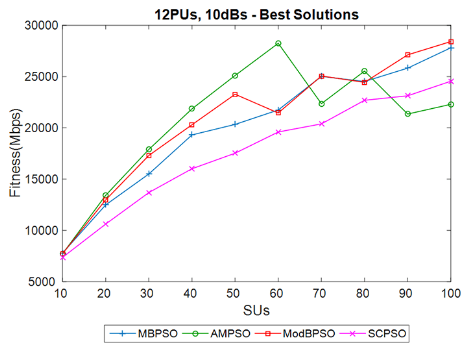
The null hypothesis (H_0) is stating no effect or no difference, whereas the alternative hypothesis (H_1) represents an effect or a difference (significant differences between algorithms) [8]. Labeling AMPSO as our control algorithm, the sign test



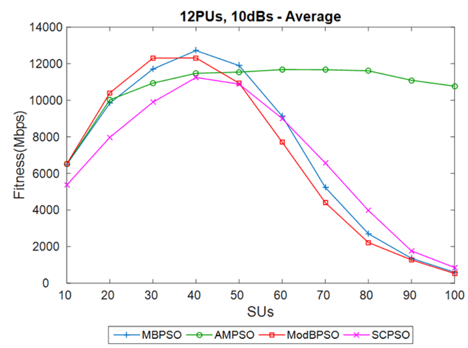
(a) best solutions γ, α = 4 dB



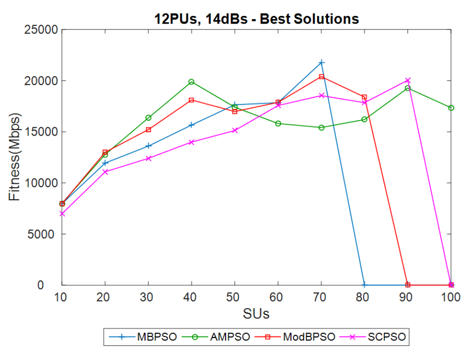
(b) average γ, α = 4 dB



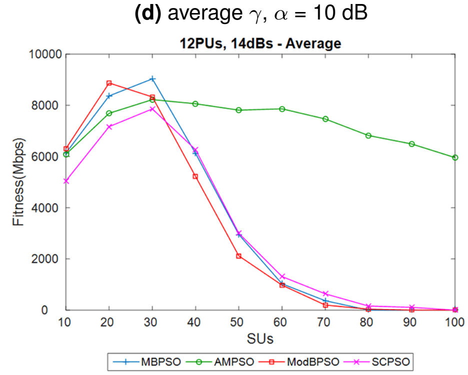
(c) best solutions γ, α = 10 dB



(d) average γ, α = 10 dB



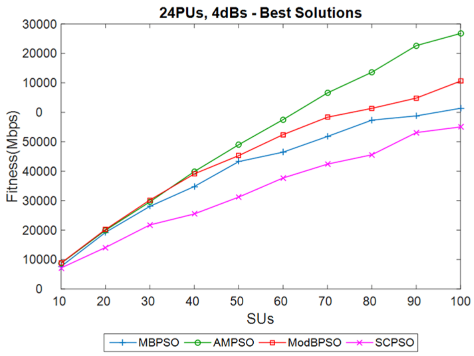
(e) best solutions γ, α = 14 dB



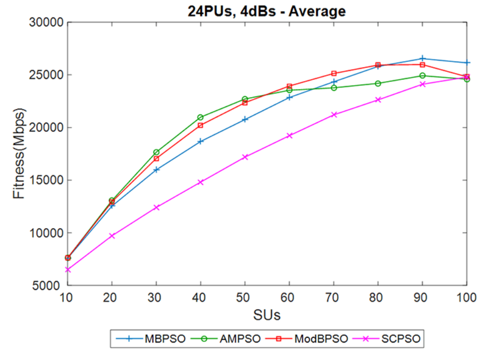
(f) average γ, α = 14 dB

Fig. 6. System throughput of the BPSO variants when Pl = 12 at γ, α = [4, 10, 14] dB

for multiple comparisons highlights those BPSO variants whose performances are statistically different when compared with the control algorithm. We apply the procedure described in [10]. Table 5 summarizes the results with two levels of significance α=0.1 and α=0.05. Let M_1 be the median response of a sample of results of the control method and M_j be the median response of a sample of results of the j-th algorithm. Let our hypotheses be H_0: $M_j \geq M_1$ and H_1: $M_j < M_1$;

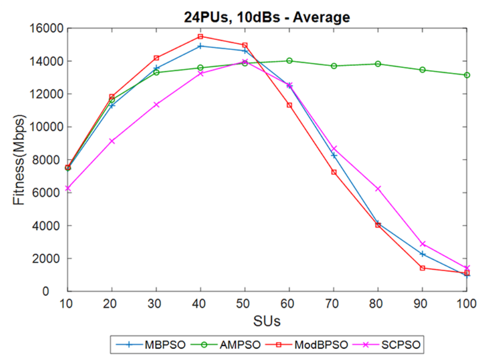


(a) best solutions γ, α = 4 dB

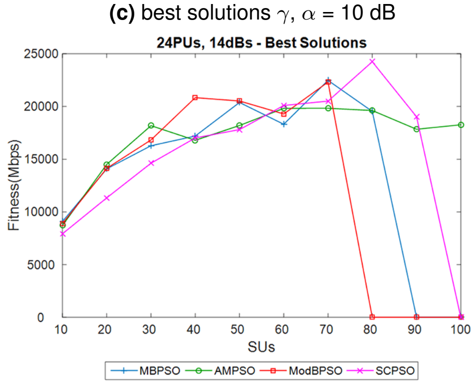


(b) average γ, α = 4 dB

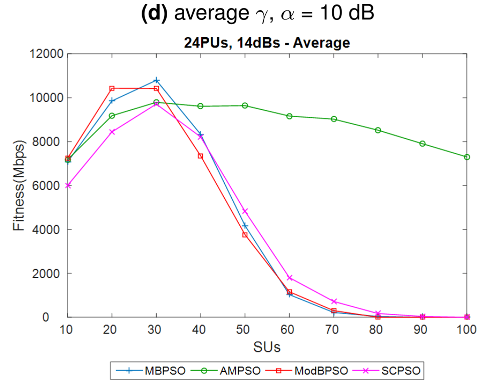
(c) best solutions γ, α = 10 dB



(d) average γ, α = 10 dB



(e) best solutions γ, α = 14 dB



(f) average γ, α = 14 dB

Fig. 7. System throughput of the BPSO variants when Pl = 24 at γ, α = [4, 10, 14] dB

that is, our control algorithm AMPSO is significantly better than the remaining algorithms. Reference to Table A.1 from [10] for (k-1)=3 and n=9 reveals that the critical values are 1 (α=0.1) and 0 (α=0.05).

Then, since the number of pluses in MBPSO, ModBPSO and SCPSO is less than or equal to the critical values, the AMPSO has a better performance than them.

Table 5. Multiple sign test using AMPSO as the control algorithm

Experiment	AMPSO 1 (control)	MBPSO 2	ModBPSO 3	SCPSO 4
6 PUs, 100 SUs, 4 dB	16610.29	12599.9 (-)	10737.3 (-)	13109.61 (-)
6 PUs, 100 SUs, 10 dB	8913.5	440.2 (-)	617.41 (-)	832.44 (-)
6 PUs, 100 SUs, 14 dB	4992.05	80.9 (-)	0 (-)	114.26 (-)
12 PUs, 100 SUs, 4 dB	20519.99	19506.12 (-)	17721.69 (-)	18941.38 (-)
12 PUs, 100 SUs, 10 dB	10769.62	594.31 (-)	524.72 (-)	852.07 (-)
12 PUs, 100 SUs, 14 dB	5957.41	0 (-)	0 (-)	0 (-)
24 PUs, 100 SUs, 4 dB	24585.18	26138.38 (+)	24820.07 (+)	24787.52 (+)
24 PUs, 100 SUs, 10 dB	13151.15	947.02 (-)	1113.33 (-)	1409.48 (-)
24 PUs, 100 SUs, 14 dB	7302.51	0 (-)	0 (-)	0 (-)
Number of pluses		1	1	1
Number of minuses		8	8	8
Critical value at α=0.1		1	1	1
Critical value at α=0.05		0	0	0

6 Discussion

Contrasting the results among the BPSO variants in Sect. 5.2 for maximum throughput, AMPSO performed better. We applied statistical tests to confirm whether AMPSO offers a significant improvement over the BPSO variants for the given experiments.

Firstly, we performed a pairwise statistical comparison using the Wilcoxon signed ranks test, confirming that AMPSO outperformed the remaining BPSO variants. In [9] is mentioned that the smaller the p-value, the stronger the evidence against H_0. In this context, we obtained small p-values (less than 0.05) when we applied the Wilcoxon signed ranks test, which indicated strong evidence against H_0.

Secondly, we performed multiple comparisons with AMPSO as the control algorithm to determine which of the other algorithms exhibit a different performance. Multiple sign test, helped us to confirm that AMPSO outperformed the BPSO variants. We used significance levels α of 0.05 and 0.1 (95% and 90% certainty that there indeed is a significant difference).

The experimental results and the nonparametric tests, confirm that in the optimization problem posed in equations (5) - (11), AMPSO produces favorable results. AMPSO is suited for complex scenarios, i.e., scenarios with high QoS requirements and many SUs and PUs deployed in the service area. Then, the non-uniform frequency distribution of binary solutions in the AMPSO search space [16] is advantageous in the SA problem due to the generating function g. As described in [16], the generating function g leads to more than one permutation of the coefficients generating the same binary solution.

They also mention that the most common solutions in the AMPSO search space are the ones that contain repetitive patterns. This trend is advantageous in problems whose optimal solutions include repetitive patterns because those solutions are common in the AMPSO search space [16]. Then in the context of the SA problem posed in equations (5) - (11), the repetitive patterns in the candidate solutions give an advantage to AMPSO.

For simple scenarios i.e., scenarios with low QoS requirements (γ, α = 4 dB), the SCPSO should be used. As more SUs are deployed in the area, it is more challenging for SCPSO, to select SUs to share a primary channel.

In contrast, MBPSO is unsuitable for complex scenarios, i.e., scenarios with high QoS requirements and many SUs. This is due to the decreasing inertia weight scheme that MBPSO uses to search for a solution. Through iterations, if fitness does not improve w increases, to stimulate exploration; otherwise, when fitness improves, w

takes a small value to exploit a region where MBPSO has found a candidate solution. However, in the binary case, as the work in [18] suggests, a smaller inertia weight enhances the exploration capability while a larger inertia weight encourages exploitation. In most cases, increasing inertia weight is favorably for the discrete PSO.

Also, from the simulation, ModBPSO had the worst performance. Although the v-shaped transfer function has been proved to have significant advantages [21], in the SA problem, it does not provide a high performance.

The objective function in (5) is the metric to measure how the HetNet efficiently uses the spectrum. Several methods exist to measure the efficiency of spectrum use, and no single measure works for all scenarios [28]. In this context from results in Sect. 5.2, maximum throughput is a well-suited metric to measure spectrum usage in scenarios with low dense cell deployments (macro and femtocells) at different QoS thresholds. Successful communications are ensured for PUs and those SUs that simultaneously exploit a channel through the QoS thresholds. Estimating the efficiency of a primary system (the set of PUs) will help to determine if it could be shared [28].

7 Conclusion and Future Work

We consider the SS paradigm in a HetNet to propose a solution to the SA problem, maximizing network throughput when one or more SUs exploit a channel simultaneously with the PU, satisfying QoS requirements on SINR. Assuming that SS will impact future next-generation cellular networks, we consider primary and secondary systems operating in that frequency band. We handle the SA problem in scenarios with high QoS requirements and many SUs and PUs deployed in an area.

Under such scenarios, the candidate solutions are high-dimensional bit strings. The search for a good solution is challenging as the QoS requirements increase. To address this challenge, we apply the AMPSO metaheuristic due to its ability to handle higher-dimensional problems. Then the AMPSO results are compared with the MBPSO, the SCPSO, and the ModBPSO.

From the simulation, AMPSO is suited for complex scenarios i.e., scenarios with high QoS requirements and a large number of SUs and PUs deployed in the service area. Then, our results confirm the AMPSO's ability to handle problems defined in larger and more abstract dimensions by combining PSO with angle modulation.

For simple scenarios i.e., scenarios with low QoS requirements (γ, α = 4 dB), the SCPSO should be used. Whereas ModBPSO is not suitable for the SA problem.

In future work, we plan to address the fairness in SUs when a channel is shared among SUs and a PU, i.e., that the SUs have the same opportunity to access spectrum to perform a communication. Also, it is planned to pose the SA problem as a multi-objective approach to maximize the data rate and the number of selected secondary links. Those objectives conflict due to the interference. Finally, we will include other components of HetNet as microcells and picocells. Those types of small cells vary in deployment location (outdoor/indoor), coverage, transmit power, and deployment configuration (planned/unplanned).

References

1. **Adedoyin, M. A., Falowo, O. E. (2020).** Combination of ultra-dense networks and other 5g enabling technologies: A survey. IEEE Access, Vol. 8, pp. 22893–22932.

2. **Andrews, J. G., Ganti, R. K., Haenggi, M., Jindal, N., Weber, S. (2010).** A primer on spatial modeling and analysis in wireless networks. IEEE Communications Magazine, Vol. 48, No. 11, pp. 156–163.

3. **Beltran, F. (2017).** Accelerating the introduction of spectrum sharing using market-based mechanisms. IEEE Communications Standards Magazine, Vol. 1, No. 3, pp. 66–72.

4. **Cave, M., Doyle, C., Webb, W. (2012).** Essentials of modern spectrum management. Cambridge University Press, New York, NY, USA.

5. **Cheng, S., Ao, W. C., Tseng, F., Chen, K. (2012).** Design and analysis of downlink spectrum sharing in two-tier cognitive femto networks. IEEE Transactions on Vehicular Technology, Vol. 61, No. 5, pp. 2194–2207.

6. **Cisco (2020).** Cisco annual internet report (2018–2023) white paper. Last accessed 12 May 2021.

7. **Deep, K., Bansal, J. C. (2008).** A socio-cognitive particle swarm optimization for multi-dimensional knapsack problem. 2008 First International Conference on Emerging Trends in Engineering and Technology, pp. 355–360.

8. **Derrac, J., García, S., Molina, D., Herrera, F. (2011).** A practical tutorial on the use of nonparametric statistical tests as a methodology for comparing evolutionary and swarm intelligence algorithms. Swarm and Evolutionary Computation, Vol. 1, No. 1, pp. 3–18.

9. **García, S., Molina, D., Lozano, M., Herrera, F. (2008).** A study on the use of non-parametric tests for analyzing the evolutionary algorithms' behaviour: a case study on the cec'2005 special session on real parameter optimization. Journal of Heuristics, Vol. 15, No. 6, pp. 617.

10. **García, S., Fernández, A., Luengo, J., Herrera, F. (2010).** Advanced nonparametric tests for multiple comparisons in the design of experiments in computational intelligence and data mining: Experimental analysis of power. Information Sciences, Vol. 180, No. 10, pp. 2044–2064. Special Issue on Intelligent Distributed Information Systems.

11. **Gupta, M. S., Kumar, K. (2019).** Progression on spectrum sensing for cognitive radio networks: A survey, classification, challenges and future research issues. Journal of Network and Computer Applications, Vol. 143, pp. 47–76.

12. **Kennedy, J., Eberhart, R. (1995).** Particle swarm optimization. Vol. 4, pp. 1942–1948.

13. **Kennedy, J., Eberhart, R. C. (1997).** A discrete binary version of the particle swarm algorithm. Vol. 5, pp. 4104–4108.

14. **Kibria, M. G., Villardi, G. P., Nguyen, K., Ishizu, K., Kojima, F. (2017).** Heterogeneous networks in shared spectrum access communications. IEEE Journal on Selected Areas in Communications, Vol. 35, No. 1, pp. 145–158.

15. **Kour, H., Jha, R. K., Jain, S. (2018).** A comprehensive survey on spectrum sharing: Architecture, energy efficiency and security issues. Journal of Network and Computer Applications, Vol. 103, pp. 29–57.

16. **Leonard, B. J., Engelbrecht, A. P. (2015).** Frequency distribution of candidate solutions in angle modulated particle swarms. 2015 IEEE Symposium Series on Computational Intelligence, pp. 251–258.

17. **Liu, C., Tsai, H. (2017).** Traffic management for heterogeneous networks with opportunistic unlicensed spectrum sharing. IEEE Transactions on Wireless Communications, Vol. 16, No. 9, pp. 5717–5731.

18. **Liu, J., Mei, Y., Li, X. (2016).** An analysis of the inertia weight parameter for binary particle swarm optimization. IEEE Transactions on Evolutionary Computation, Vol. 20, No. 5, pp. 666–681.

19. **Martinez, E., Andrade, A. G., Martínez-Vargas, A., Galaviz, G. (2016).** Optimización binaria por cúmulo de partículas con memoria (MBPSO) para resolver un problema de espectro compartido. Computación y Sistemas, Vol. 20, pp. 153–168.

20. **Martínez-Vargas, A., Andrade, Á. G., Sepúlveda, R., Montiel-Ross, O. (2014).** An admission control and channel allocation algorithm based on particle swarm optimization for cognitive cellular networks. In **Castillo, O., Melin, P., Pedrycz, W., Kacprzyk, J.**, editors, Recent Advances on Hybrid Approaches for Designing Intelligent Systems. Springer International Publishing, Cham, pp. 151–162.

21. **Mirjalili, S., Lewis, A. (2013).** S-shaped versus v-shaped transfer functions for binary particle swarm optimization. Swarm and Evolutionary Computation, Vol. 9, pp. 1–14.

22. **Olivas, F., Valdez, F., Melin, P., Sombra, A., Castillo, O. (2019).** Interval type-2 fuzzy logic for dynamic parameter adaptation in a modified gravitational search algorithm. Information Sciences, Vol. 476, pp. 159–175.

23. **Pampara, G., Franken, N., Engelbrecht, A. P. (2005).** Combining particle swarm optimisation with angle modulation to solve binary problems. Vol. 1, pp. 89–96.

24. **Peha, J. M. (2005).** Approaches to spectrum sharing. IEEE Communications Magazine, Vol. 43, No. 2, pp. 10–12.

25. **Roeva, O., Zoteva, D., Castillo, O. (2021).** Joint set-up of parameters in genetic algorithms and the artificial bee colony algorithm: an approach for

cultivation process modelling. Soft Comput., Vol. 25, No. 3, pp. 2015–2038.

26. **Rohoden, K., Estrada, R., Otrok, H., Dziong, Z. (2020).** Evolutionary game theoretical model for stable femtocells' clusters formation in hetnets. Computer Communications, Vol. 161, pp. 266–278.

27. **Ruby, D., Vijayalakshmi, M., Kannan, A. (2019).** Intelligent relay selection and spectrum sharing techniques for cognitive radio networks. Cluster Computing, Vol. 22, No. 5, pp. 10537–10548.

28. **Rysavy, P. (2014).** Challenges and considerations in defining spectrum efficiency. Proceedings of the IEEE, Vol. 102, No. 3, pp. 386–392.

29. **Sathya, V., Kala, S. M., Bhupeshra, S., Tamma, B. R. (2021).** Raptap: a socio-inspired approach to resource allocation and interference management in dense small cells. Wireless Networks, Vol. 27, No. 1, pp. 441–464.

30. **Talbi, E. G. (2009).** Metaheuristics: From Design to Implementation. Wiley Publishing.

31. **Talukdar, B., Kumar, D., Hoque, S., Arif, W. (2021).** Cooperative spectrum sensing in energy harvesting cognitive radio networks under diverse distribution models. In **Mandloi, M., Gurjar, D., Pattanayak, P., Nguyen, H.**, editors, 5G and Beyond Wireless Systems: PHY Layer Perspective. Springer Singapore, Singapore, pp. 245–272.

32. **Tragos, E. Z., Zeadally, S., Fragkiadakis, A. G., Siris, V. A. (2013).** Spectrum assignment in cognitive radio networks: A comprehensive survey. IEEE Communications Surveys Tutorials, Vol. 15, No. 3, pp. 1108–1135.

33. **Valdez, F., Vazquez, J. C., Melin, P., Castillo, O. (2017).** Comparative study of the use of fuzzy logic in improving particle swarm optimization variants for mathematical functions using co-evolution. Applied Soft Computing, Vol. 52, pp. 1070–1083.

34. **Yang, J., Zhang, H., Ling, Y., Pan, C., Sun, W. (2014).** Task allocation for wireless sensor network using modified binary particle swarm optimization. IEEE Sensors Journal, Vol. 14, No. 3, pp. 882–892.

35. **Zhen Ji, Tao Tian, Shan He, Zexuan Zhu (2012).** A memory binary particle swarm optimization. 2012 IEEE Congress on Evolutionary Computation, pp. 1–5.

36. **Zia, K., Javed, N., Sial, M. N., Ahmed, S., Pirzada, A. A., Pervez, F. (2019).** A distributed multi-agent rl-based autonomous spectrum allocation scheme in d2d enabled multi-tier hetnets. IEEE Access, Vol. 7, pp. 6733–6745.

Article received on 29/06/2021; accepted on 16/11/2021.
Corresponding author is Anabel Martínez-Vargas.