Comparison of Hybrid Firefly Algorithms for Binary and Continuous Optimization Problems in a TV White Space Network

RONOH KENNEDY [1]*, KAMUCHA GEORGE [2]

[1]Department of Computer Science, Technical University of Kenya, Nairobi, Nairobi, KENYA.
[2]Department of Electrical and Information Engineering, University of Nairobi, Nairobi, KENYA

Abstract: - TV white spaces (TVWS) can be used by Secondary Users (SUs) through Dynamic Spectrum Access (DSA) as long as they do not cause harmful interference to Primary Users (PUs). Due to spectrum scarcity, there is increasing demand for DSA. When there is a high density of SUs in a TVWS network such as cellular access to TVWS, problem of interference among SUs will arise. Possibility of harmful interference to PUs may also arise. Power and spectrum allocation optimization is therefore necessary to reduce the level of interference among SUs and to protect PUs against harmful interference. In this paper different hybrid firefly algorithms with particle swarm optimization and genetic algorithm for optimization of spectrum allocation in a TVWS network as a discrete optimization problem and that of power allocation as a continuous optimization problem are compared. Simulation was done using Matlab. Simulation results show that hybrid firefly algorithm with genetic algorithm outperforms other hybrid firefly algorithms for spectrum allocation. On the other hand, hybrid firefly algorithm with genetic algorithm and particle swarm optimization outperforms all other algorithms for power allocation as continuous optimization problem.

1 Introduction

Optimization algorithms can be classified as either exact or approximate. Exact optimization algorithm guarantees that the optimum solution will be found. Exact methods are also known as traditional optimization methods. An example of exact optimization is brute force. Brute force tries to find every possible solution in search space so that the global optimal solution is guaranteed to be found. As the search space increases, the time and space complexity for an exact algorithm increases. Exact algorithms are not suitable for NP hard optimization problems. This is because the time complexity of NP hard problem increases with problem size. Other examples of exact algorithms are Simplex method of linear programming, divide and conquer and dynamic programming. The main disadvantage of exact algorithms or traditional algorithms is that they have poor time complexity [1]. Approximate methods, on the other hand, are used to find near-optimal solutions for NP-hard problems in polynomial time. Approximate methods do not guarantee global optimal solution but they provide a near optimal solution. They are applied for solving intractable combinatorial optimization problems because they give a good solution quality with reasonable time complexity.

Approximate optimization algorithms are generally referred to as heuristics and are classified into two: heuristic and metaheuristic. Heuristic algorithms are specific to a problem because one heuristic solution to an optimization problem cannot be applied to another optimization problem. A metaheuristic algorithm, on the other hand, is problem independent. Metaheuristic algorithms provide guidelines or a series of steps that can be applied to different optimization problems.

Among other heuristic algorithms, population based metaheuristic algorithms are preferred for NP hard optimization problems. This because they have better ability for global exploration and local exploitation in searching the solution space in addition to having reasonable time complexity [2]. Population-based metaheuristic algorithms perform optimization as follows: a population of individuals is initialized whereby each individual represents a possible solution to the optimization problem. The population of individual solutions is continuously improved through a number of iterations. The best

solution at the end of iterations cycle represents the solution to the optimization problem.

Population-based metaheuristic algorithms can be classified as either swarm intelligence algorithms or evolutionary algorithms. Evolutionary algorithms mimic behavior of biological entities and evolution and are also inspired by Darwin's theory of evolution [3]. An example of evolutionary algorithm is Genetic Algorithm (GA). Swarm intelligence algorithms mimic the behavior of a swarm of biological entities. Examples of swarm intelligence algorithms include Particle Swarm Optimization (PSO), Firefly Algorithm (FA), Artificial Bee Colony (ABC) and Ant Colony Optimization (ACO).

Despite the advantages of evolutionary algorithms and swarm intelligence algorithms, they can get trapped in a local optimum [4]. Recent trend has been to hybridize Population-based metaheuristic algorithms so as to overcome the shortcoming by improving either the exploration or exploitation ability or both [4], [5].

The objective of this paper is to compare the performance of different hybrid firefly algorithms (FA) for spectrum allocation in a TV White Space (TVWS) network as discrete or binary optimization problem and that of power allocation as a continuous optimization problem. This is because the performance of population based metaheuristic algorithms depends on the nature of the optimization problem. This also applies to hybrid population based metaheuristic algorithms. The algorithms to be compared are hybrid FA with genetic algorithm (GA), hybrid FA with particle swarm optimization (PSO) as well as hybrid FA, GA and PSO. Simulation results show that hybrid FA with GA results in better spectrum allocation as measured by SU sum throughput. On the other hand, hybrid FA with GA and PSO results in better power allocation as measured by sum power and sum throughput.

The rest of the paper is organized as follows. Section 2 provides a review of related work. System model and the algorithms are presented in section 3. Section 4 presents the simulation set up. Performance evaluation of the algorithms is discussed in section 5. Analysis and comparison of hybrid FA algorithms for spectrum and power allocation is discussed in section 6. The paper is concluded in section 7.

2 Related Work

In this section, related work on hybrid power and spectrum allocation in TVWS network as well as hybrid FA algorithms has been presented.

2.1 Spectrum Allocation Using Population-Based Metaheuristic Algorithms

Spectrum allocation using firefly algorithm has been explored by Anumandla et al. in [6] and Liu t al. in [7]. The results of the two papers show that FA gives a better solution and converges to a solution faster than GA and PSO. For this reason FA is chosen as the base algorithm.

A summary of the use of evolutionary algorithms for spectrum allocation in cognitive radio networks has been presented by Zhao et al in [8]. The work discusses the use particle swarm optimization, genetic algorithm and quantum genetic algorithm. The work finds that PSO converges much faster and gives a better solution compared to GA and Color Sensitive Graph Coloring (CSGC). Performance of CSGC is found to be lower than both PSO and GA in terms of solution quality.

Spectrum allocation has been abstracted as a graph coloring problem by Elhachmi and Guennounin [9]. GA is then used to find the best spectrum allocation matrix. In this research we assume that each SU interferes with every other SU in the network since a small network is considered. Graph coloring will, therefore, not be used to represent interference constraints.

A spectrum allocation framework based on PSO and simulated annealing has been presented by Jie and Tiejun in [10]. Simulated annealing is introduced to prevent prematurity of particle swarm optimization. The work finds that PSO with simulated annealing performs better than graph coloring and greedy algorithms in terms of solution quality.

Spectrum allocation using graph coloring and ant colony optimization (ACO) has been presented Koroupi et al. in [11]. ACO is found to perform better than PSO and CSGC in terms of solution quality. However, the run time is higher than that of PSO.

Our previous work in [12] presents the used of modified firefly algorithm for joint power and spectrum allocation for a TVWS network. FA is modified to solve a continuous-discrete optimization problem. In this work, we compare the performance of hybrid algorithms for spectrum allocation as a discrete optimization problem.

2.2 Power Allocation Using Population-Based Metaheuristic Algorithms

In [13], we proposed a firefly algorithm based power allocation algorithm in a TVWS network which makes use of a Geo-location Database (GLDB) and that considers interference constraints at both PU and SUs. To ensure adequate protection of PUs, both co-channel and adjacent channel interference are considered. In this paper, we compare the performance of the proposed algorithm in the paper with other hybrid FA algorithms.

In [14], we proposed a hybrid FA, GA and PSO (FAGAPSO) algorithm for power allocation in TVWS network. The results show that FAGAPSO performs far much better than FA, GA and PSO. In this paper the performance of different hybrid FA, GA and PSO are compared for power allocation as a continuous optimization problem and spectrum allocation as a binary or discrete optimization problem.

2.3 Hybrid Firefly Algorithm and Particle Swarm Optimization

Arunacham et. al. [15] proposed a hybrid FA and PSO for problem of combined economic and emission dispatch including valve point effect. In the proposed algorithm, there is no modification to firefly algorithm but the initial solution is obtained from PSO. The authors argue that quality of the final solution of FA depends on the initial solution. Simulation results show that the hybrid the algorithm performs better than both PSO and FA. This algorithm was applied to a continuous optimization problem. There is need to investigate the performance for a binary optimization problem.

Kora P. and Krishna K. [16] also proposed a hybrid FA and PSO algorithm for detection of bundle branch block. The hybrid algorithm makes use of PSO concepts and parameters. The concepts of personal best and global best which are absent in FA are introduced. All the steps in FA remain the same with that of the proposed algorithm except that equation of FA that represents firefly movement is changed to incorporate the idea of personal best and global best. In the proposed algorithm, each firefly movement involves a move towards the local best and global best. This algorithm was applied to a continuous optimization problem. There is need to investigate the performance for binary optimization problem.

2.4 Hybrid Firefly Algorithm and Genetic Algorithm

Rahmani A. and Mirhassani S.A. [17] proposed a hybrid FA and GA. All the steps in the FA remain the same except that for every iteration, the two current best solutions are crossed over. Two fittest offsprings out of the four offsprings are then selected. For mutation, one of the two offsprings is randomly selected. If the selected offspring has a better solution compared to the current best solution, it replaces the current best solution. This algorithm was applied to a binary optimization problem. There is need to investigate the performance for continuous optimization problem.

Luthra J. and Pal Saibal K. [18] also proposed a hybrid FA and GA for the solution of the monoalphabetic substitution cipher. In the proposed algorithm, movement of fireflies in space is done using genetic operators and the concept of dominant gene cross over. With dominant gene cross over, an offspring takes more from one parent than the other during cross over. This algorithm was applied to a binary optimization problem. There is need to investigate performance for a continuous optimization problem.

2.5 Hybrid Firefly Algorithm Particle Swarm Optimization and Genetic Algorithm

Our previous work in [19] presents the use of hybrid FA, PSO and GA for joint power and spectrum allocation for a TVWS network. In this work, we compare the performance of a variety of hybrid FA, GA and PSO algorithms for spectrum allocation only as a discrete optimization problem and that of power allocation as a continuous optimization problem.

2.6 Other Hybrid Firefly Algorithms

There also exists other hybrid FA with other population based metaheuristic algorithms. A hybrid of bat algorithm and FA has been proposed by Warangal et al. in [20]. A hybrid of cuckoo search of FA has been proposed by Elkhechafi et al. in [21]. A hybrid of FA and ant colony optimization has been proposed by Layeb and Benayad in [22]. PSO and GA are chosen to be hybridized with FA because they are able to converge to a good solution more quickly compared to other population based metaheuristic algorithms

3. System Model and Resource Allocation Algorithms

In order to improve QoS in a TVWS network, there is need to optimize resource (power or spectrum) allocation to Secondary Users (SUs). In a network where there is high number of devices

seeking access to a secondary network allocation of two resources, power and spectrum, need to be optimized to ensure that as many SUs as possible access the secondary network while ensuring that interference constraints for PUs and QoS requirements for SUs are met [19], [12]. In this paper, resource allocation refers to allocation of power and spectrum to SUs.

Resource allocation in a TVWS network is a NP hard optimization problem. Hence metaheuristic algorithms are preferred over exact algorithms. Among other population-based metaheuristic algorithms, FA is chosen for resource allocation because it has been found to perform better than other algorithms in terms of solution quality and convergence time [23] [24]. Despite its superior performance over other algorithms, FA can get trapped in local optimum. Hybridizing FA with other algorithms enables FA to avoid being trapped in a local optimum.

FA is hybridized with PSO and GA in order to improve quality of solution obtained by FA through improvement of its exploitation or exploration ability. PSO is chosen because, compared to other EAs, it has been found to converge more quickly and give better quality solutions [25]. Crossover feature of GA can be further added to the hybrid of FA and PSO to further improve the quality of the solution through diversification of the search of the solution space and hence avoidance of the problem FA of being trapped into local optimum.

A network illustrated by Figure 1 is considered. In the figure there is a single TV receiver placed at the edge of the protection region. Among all the TV receivers in the protection region, a TV receiver at this location is the one which is most vulnerable to interference since it is very close to the secondary network and receives the weakest from the TV tower. GLDB regulations require that the D/U ratio or protection ratio be measured at the edge of protection region [26]. Aggregate interference at the TV receiver, both co-channel and adjacent channels should not make the protection ratio fall below the required protection ratio threshold.

3.1 Spectrum Allocation Using Binary Firefly Algorithm

Optimization of spectrum allocation only using FA is first considered. In order to reduce interference to the PU and among SUs, optimization of spectrum allocation is necessary. Let the potential channel allocation matrix be represented as $A = \{a_{n,m} \| a_{n,m} \in \{0,1\}$. A is of dimension N×M, where N is the number of SUs and M is the number of channels. $a_{n,m} = 1$ if channel m assigned to user n. $a_{n,m} = 0$ if channel m is not assigned to user n. Spectrum allocation, therefore, deals with binary values unlike for the case of power allocation where the values are continuous. The optimization goal is to find a channel allocation matrix A^* to maximize sum throughput of all SUs defined in equation 1.
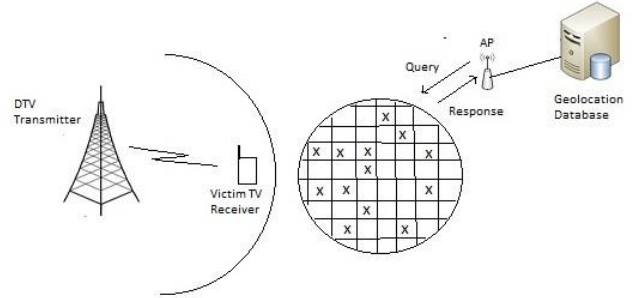

Figure 1. Interference scenario.

$$U = \sum_{n=1}^{N} \sum_{m=1}^{M} r_{n,m}, \tag{1}$$

where U is the sum throughput, b_m is the bandwidth of channel m and $r_{n,m}$ is the throughput of single SU transmitter n on channel m computed as follows:

$$r_{n,m} = \frac{1}{2} a_{n,m} b_m log_2(1 + \rho_n), \tag{2}$$

where the term ρ_n is SU SINR [12]. Optimal channel allocation matrix, A^*, can be found by solving the optimization Problem 1.

Problem 1

$$A^* = \arg max \, U \tag{3}$$

subject to:

$$C_1: \omega > \omega_o, \tag{4}$$
$$C_2: \rho_n > \rho_o, n = 1,2 \dots . N, \tag{5}$$
$$C_3: a_{n,m} \in \{0,1\}, \tag{6}$$
$$C_4: a_{n,m} = 1, c_n = m, \tag{7}$$
$$C_5: a_{n,m} = 0, c_n \neq m, \tag{8}$$

where is ρ_n SU SINR for SU n, ρ_o is the minimum required SU SINR, ω_o is the minimum required PU SINR [12]. Constraints C_3 , C_4 and C_5 imply that one

Algorithm 2: Optimization of Spectrum Allocation using FAGAPSO

Step 1:
- 1.1 Initialize number of particles, c_1, c_2, ω, v_{min}, v_{max}
- 1.2 Assign each SU randomly power values within range.
- 1.3 For each particle

  Initialize particle with random channel assignment

  End
- 1.4 Do

  1.4.1 For each particle

  Calculate fitness value using equation (1)

  If the fitness value is better than the best fitness value (p_i) in history

  set current value as the new p_i

  End

  1.4.2 Choose the particle with the best fitness value of all the particles as the p_{best}

  If current p_{best} and its associated x_{best} is better than g_{best} set current p_{best} as g_{best}

  1.4.3 For each particle

  Calculate particle velocity according equation (20)

  Update particle position according equation (21) and with application of

  equations (18) and (19)

  If particle position update result in allocation of more than one channel,

  randomly select only one channel

  End

  While maximum iterations has not been reached.
- 1.5 g_{best} set as the final solution of PSO.

Step 2
- 2.1 Initialize the control parameters of the algorithm α, β, γ firefly number NP and maximum number of iterations t_{max}.
- 2.2 Set the dimension of fireflies D.
- 2.3 Set initial position of fireflies as those of the solution for Problem 3.4 generated by PSO in Step 1.

Step 3
- 3.1 Calculate the fitness value of each firefly using equation (1) and rank the fireflies according to their fitness values.
- 3.2 Find the current best solution.
- 3.3 Apply crossover mechanism on the top two best solutions.
- 3.4 Select the best offspring out of the four offsprings created through crossover and use it as the current best solution of FA if its fitness is better than that of the current best.

Step 4
- 4.1 For every firefly, move it to the better solution according to equation (22) and with application of equations (16), (17) and (18).
- 4.2 Check firefly x_i to see if each firefly has only channel assignment to SU. If there is an SU that has been assigned more than one channel, randomly select one channel one only.

Step 5
- If it reaches the predefined maximum number of iterations, then the power vector of the current best solution mentioned in step 3 is derived and stop the progress else go to step 3 and continue.

channel $(c_n = m)$ only in SU_n channel allocation row will have a value 1, the rest will be 0.

The optimization of spectrum allocation problem defined in Problem 1 is a constrained optimization problem. The most common way to deal with constraints when using evolutionary algorithms to solve optimization problems is to use an exterior penalty function [27]. Exterior penalty functions are preferred over interior penalty functions because they do not require an initial feasible solution [27]. Penalty functions changes a constrained optimization problem into an unconstrained optimization problem. This is achieved by adding to the objective function, a penalty term that prescribes a high cost for violation of constraints. By applying penalty functions, the objective function of optimization Problem 1 will change to:

$$\emptyset(A) = U - c_s \sum_{n=1}^{N} \max[0, g_n^s]^2 - c_p \max[0, g_n^p]^2, \qquad (9)$$

where $g_n^s = \rho_o - \rho_n$ and $g_n^p = \omega_o - \omega_n$. The terms c_s and c_p are penalty factors for SU SINR threshold violation and PU SINR threshold violation respectively, respectively. The optimization Problem 1 can then be re-written as:

Problem 2
$$A^* = \arg\,max\,\emptyset(A) \qquad (10)$$
subject to:
$$C_1: \omega > \omega_o, \qquad (11)$$
$$C_2: \rho_n > \rho_o, n = 1,2 \dots\dots N, \qquad (12)$$
$$C_3: a_{n,m} \in \{0,1\}, \qquad (13)$$
$$C_4: a_{n,m} = 1, c_n = m, \qquad (14)$$
$$C_5: a_{n,m} = 0, c_n \neq m. \qquad (15)$$

Algorithm 1 shows the steps for solving Problem 2 using FA. The algorithms starts with specifying the number of fireflies as NP and dimension of each firefly as $D = N$. For spectrum allocation, each firefly represents a potential solution to the problem of finding optimal spectrum allocation to all SUs in the TVWS network. Each firefly consists of channel allocation matrices. Each firefly will have the same power allocation assigned to SUs that is randomly generated. In all the steps, power assignment will not change. At every iteration, the best firefly is determined and each firefly movement is done according to Step 3 in Algorithm 1 using equation (16).

$$x_i^{t+1} = x_i^t + \beta_o e^{-\gamma r_{ij}^2}(x_j^t - x_i^t) + \alpha_t \epsilon_t^i, \quad (1\,6)$$

where terms x_i and x_j are the locations of firefly i and firefly j, β_o denotes the light intensity of the source, r is the distance between two fireflies, γ is the light absorption co-efficient, the symbol α is randomization parameter and the term ϵ_t^i is a vector of random numbers with uniform distribution. The second term represents attractiveness while the third term represents randomization. The symbol t is the iteration number. The following equation will be used for computing distance (r_{ij}) between two channel allocation matrices:

$$r_{ij} = \sum_{d=1}^{D} \sum_{m=1}^{M} x_{m,d,i} \otimes x_{m,d,j}, \qquad (17)$$

where $x_{m,d,i}$ and $x_{m,d,j}$ are the channel allocation values in fireflies i and j, respectively at position m, d in the channel allocation matrix. In order to determine whether $x_{m,d,i}$ will be a 0 or 1, Sigmoid function is first used to change the value after the firefly mobility by the

$$sig(x_{m,d,i}) = \frac{1}{1 + e^{-x_{m,d,i}}}. \qquad (18)$$

The value for each position in the channel allocation matrix is then computed as follows:

$$x_{m,d,i}^{t+1} = \begin{cases} 1 & rand() < sig(f) \\ 0 & else \end{cases} \quad whe\text{re } f$$
$$= x_{m,d,i}^t. \qquad (19)$$

After a fixed number of iterations, the firefly with the best objective function with the associated spectrum allocation matrix is

selected as the solution to the spectrum allocation problem.

3.2 Spectrum Allocation Using Binary Hybrid Firefly and Particle Swarm Optimization with Genetic Operators

In this section, optimization of spectrum allocation using binary hybrid firefly and particle swarm optimization with genetic operators (FAGAPSO) is discussed. The algorithm steps are outlined in Algorithm 2. In step 1 of Algorithm 2, optimization spectrum allocation is first done using PSO. Each PSO particle is made up channel allocation matrix of dimension N×M. In PSO, each particle looks at three parameters: its current position (X_i), best position (P_i), and its flying velocity (V_i). For every iteration, the best particle and its position P_g are determined. Each particle then flies towards the best particle with its current velocity. Each particle updates its current velocity according to the equation (20):

$$New\ V_i\ =\ \omega\ \times current\ V_i\ +\ \ c_1 \times\ rand()\ \times\ (P_i - X_i)\ +\ \ c_1 \times rand() \times (P_g\ -\ X_i) \qquad (\ 20)$$

where c_1 and c_1 are two positive constants named, rand() is a random function, V_{max} is the maximum particle velocity. w plays the role of balancing local search and global search. With the new current velocity, the position of the particle is then updated according to the equation (21):

$$New\ position\ X_i \\ =\ current\ position\ X_i \\ +\ New\ V_i \qquad (21)$$

$$V_{max} \geq V_i \geq -V_{max}$$

Position update in step 1.4.3 of Algorithm 1 may result in values that are not binary. Just like FA, equations (18) and (19) will be used to convert the values obtained during particle position update to binary values. If particle position update result in allocation of more than one channel to SU, one channel will be randomly.

In step 2, FA starts with initial solution of PSO generated in Step 1. All fireflies will be initiated with solutions found in PSO particles at the end of PSO in Step 1. Each firefly is made up channel allocation matrix of dimension N×M. In step 3, after ranking fireflies according to their fitness, the best two fireflies are crossed over to generate four new offsprings. The four new offsprings are then ranked according to their fitness. Single point cross over is used. The current best firefly will then be replaced by the best offspring if its fitness is lower (better) than that of the best offspring. Instead of firefly movement being that described by equation (16), firefly movement will involve local search towards local personal best and global best according to equation (22). The algorithm therefore makes use some PSO operators including P_{best}, Pg_{best}, c_1 and c_2 according to equation (22).

$$x_i^{t+1} \\ = x_i^t + c_1 e^{-\gamma r_{ij}^2}(p_i - x_i^t) + c_2 e^{-\gamma r_{ij}^2}(p_g - x_i^t) \\ + \alpha_t \epsilon_t^i \qquad (22)$$

3.3 Power Allocation Using Continuous Hybrid Firefly and Particle Swarm Optimization with Genetic Operators

The optimization problem is as follows [13]:

Problem 3

$$p^* = \arg min\ \emptyset(p) \qquad (23)$$

subject to $C: p_{min} \leq\ p_i \leq\ \ p_{max}$

where

$$\emptyset(p) =\ \varphi(p) + c_s \sum_{i=1}^{N} \max[0, g_i^s]^2 + \\ c_p \max[0, g_i^p]^2. \quad (24)$$

The optimization problem in Problem 3 is about minimization of sum power and minimization of interference threshold violations at SUs and at the PU. In equation (23), the first term, $\varphi(p)$, represents the sum power of all SUs, the second term $(c_s \sum_{i=1}^{N} \max[0, g_i^s]^2)$ represents interference threshold violation for SUs while the third term represents interference threshold

Ronoh Kennedy, Kamucha George

Algorithm 2: Optimization of Spectrum Allocation using FAGAPSO

Step 1
- 1.1 Initialize number of particles, $c_1, c_2, \omega, v_{min}, v_{max}$
- 1.2 Assign each SU randomly power values within range.
- 1.3 For each particle

    Initialize particle with random channel assignment

    End
- 1.4 Do

    1.4.1 For each particle

        Calculate fitness value using equation (1)

        If the fitness value is better than the best fitness value (p_i) in history

        set current value as the new p_i

    End

    1.4.2 Choose the particle with the best fitness value of all the particles as the p_{best}

        If current p_{best} and its associated x_{best} is better than g_{best} set current p_{best} as

        g_{best}

    1.4.3 For each particle

        Calculate particle velocity according equation (20)

        Update particle position according equation (21) and with application of

        equations (18) and (19)

        If particle position update result in allocation of more than one channel,

        randomly select only one channel

    End

    While maximum iterations has not been reached.
- 1.5 g_{best} set as the final solution of PSO.

Step 2
- 2.1 Initialize the control parameters of the algorithm α, β, γ firefly number NP and maximum number of iterations t_{max}.
- 2.2 Set the dimension of fireflies D.
- 2.3 Set initial position of fireflies as those of the solution for Problem 3.4 generated by PSO in Step 1.

Step 3
- 3.1 Calculate the fitness value of each firefly using equation (1) and rank the fireflies according to their fitness values.
- 3.2 Find the current best solution.
- 3.3 Apply crossover mechanism on the top two best solutions.
- 3.4 Select the best offspring out of the four offsprings created through crossover and use it as the current best solution of FA if its fitness is better than that of the current best.

Step 4
- 4.1 For every firefly, move it to the better solution according to equation (22) and with application of equations (16), (17) and (18).
- 4.2 Check firefly x_i to see if each firefly has only channel assignment to SU. If there is an SU that has been assigned more than one channel, randomly select one channel one only.

Step 5
- If it reaches the predefined maximum number of iterations, then the power vector of the current best solution mentioned in step 3 is derived and stop the progress else go to step 3 and continue.

violation for PU. The terms c_s and c_p are penalty factors for SU interference threshold violation and PU interference threshold violation.

The algorithm steps are outlined in Algorithm 3. In step 1 of Algorithm 3, optimal power allocation is first found using PSO. In step 2, FA starts with initial solution of PSO generated in Step 1. All fireflies will be initiated with solutions found in PSO particles at the end of PSO in Step 1. In step 3, after ranking fireflies according to their fitness, the best two fireflies are crossed over to generate four new offsprings. The four new offsprings are then ranked according to their fitness. The current best firefly will then be replaced by the best offspring if its fitness as measured by equation (23) is lower (better) than that of the best offspring. Firefly movement will involve a search towards local personal best and global best according to equation (25). The proposed algorithm therefore makes use of some PSO operators including P_{best}, g_{best}, c_1 and c_2.

$$x_i^{t+1} = x_i^t + c_1 e^{-\gamma r_{ij}^2}(p_i - x_i^t) + c_2 e^{-\gamma r_{ij}^2}(p_g - x_i^t) + \alpha_t \epsilon_t^i, \quad (25)$$

4. Simulation Setup

Parameters used in the simulation are outlined in Table 1. Simulation was done using Matlab R2016a. Matlab is chosen because it is rich in in-built functions. SUs (N=200, 400, 600, 800, 1000) are distributed over an area of 1. Figure 2 shows the network diagram generated in Matlab. Initially SUs are distributed randomly across channels varying from 2 to 10 in steps of 2 i.e M =2, 4, 6,8, and 10. Parameters used for FA were as follows: $\beta_o = 1$, $\alpha = 30$, $\gamma = 10$, number of fireflies NP = 20. Parameters used for PSO were as follows: number of particles = 20, inertia weights: $w_{max} = 4$, $w_{min} = 2$, social parameter $c_1 = 2$ and cognitive parameter $c_2 = 2$. Parameters used for GA were as follows: number of chromosomes=20, mutation rate = 0.8 and selection rate = 0.5. The parameters were set using trial and error and also consideration of values used in a number of peer reviewed journals. Different parameter values were tried before settling on the stated parameters that provided good performance.
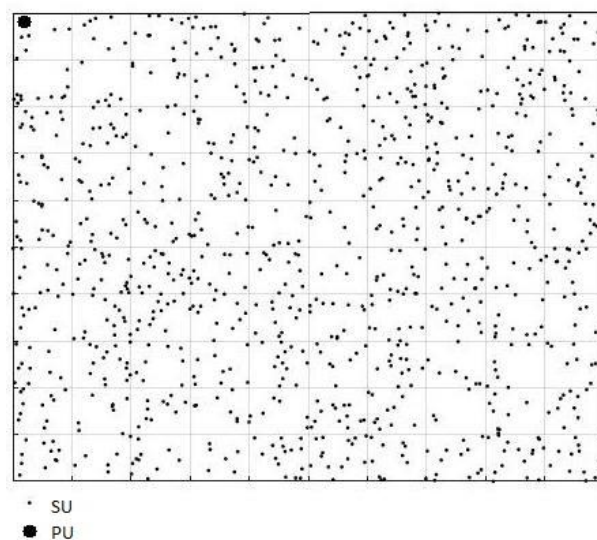

Figure 2. Network diagram

5. Performance Evaluation and Analysis

This section presents and discusses simulations results for both spectrum allocation and power allocation.

5.1 Spectrum Allocation

In this section, simulation results for optimization of spectrum allocation using a variety of hybrid FA, PSO and GA algorithms are presented. FAGAPSO is compared with:

- FA
- GA
- PSO
- FAPSO1: FA with initial solution of PSO
- FAPSO2: FA with PSO operators i.e firefly movement using P_{best}, g_{best}, c_1 and c_2 as expressed in equation 22
- FAPSO3: FA with PSO operators with initial solution of PSO
- FAGA: Firefly algorithm with the use crossover feature of GA according to Step 3.3 of Algorithm 2.

Algorithm 3: Optimization of Power Allocation using FAGAPSO

Step 1
- 1.1 Initialize number of particles, $c_1, c_2, \omega, v_{min}, v_{max}$
- 1.2 For each particle

    Initialize particle with random power values that are within allowed range.

    End
- 1.3 Do

    For each particle

      Calculate fitness value using equation (23)

      If the fitness value is better than the best fitness value (p_i) in history

      set current value as the new p_i

    End

    1.4 Choose the particle with the best fitness value of all the particles as the p_{best}
- 1.5 If current p_{best} is better than g_{best} set current p_{best} as g_{best}

    1.6 For each particle

      Calculate particle velocity

      Update particle position

    End

    1.7 While maximum iterations has not been reached.
- 1.8 Set g_{best} as the final solution of PSO.

Step 2
- 2.1 Initialize the control parameters of the algorithm α, β, γ firefly number NP and maximum number of iterations t_{max}.
- 2.2 Set the dimension of fireflies D.
- 2.3 Set initial position of fireflies as those of the solution for Problem 1 generated by PSO in Step 1.

Step 3
- 2.4 Calculate the fitness value of each firefly using equation (23) and rank the fireflies   according to their fitness values.
- 2.4 Find the current best solution.
- 2.5 Apply crossover mechanism on the top two best solutions.
- 2.6 Select the best offspring out of the four offsprings created through crossover and use it as the current best solution of FA if its fitness is better than that of the current best.

Step 4
- 2.7 For every firefly, move it to the better solution according to equation (1).
- 2.8 Check firefly x_i to see if the all the power values in the power vector are within range. If any values are out of range then create random values that are within range to replace them.

Step 5
- 2.9 If it reaches the predefined maximum number of iterations, then the power vector of the current best solution mentioned in step 3 is derived and stop the progress else go to step 3 and continue.

100 iterations were used for GA, FA, PSO, FAGA and FAPSO2. In FAPSO1, FAPSO3 and FAGAPSO, 50 iterations were used for both FA and PSO so that the total number of iterations will be also 100. Simulation results are generated for 10 simulation runs and an average is done. The performance of the algorithms is compared using the following metrics: objective function value and sum throughput.

5.1.1 Objective Function Value

Figure 3 shows comparison of FAGAPSO with the rest of the algorithms in terms of achieved objective function value (equation (9)) for different values of N and M set to 10. The results show that FAGA achieves the best (highest) objective function (equation (9)) value compared to all other algorithms for all values of N. The objective function value increases as N is increased from 200 to 400 for all the algorithms. However, after a value of N=400, the objective function value generally flattens. The flattening is because the increase in objective function value is offset by the penalty function terms in equation (9). At N=1000, the objective function value starts to reduce

because the penalty function terms value starts becoming more significant such that it reduces the sum throughput in equation (9).

5.1.2 Sum Throughput

Figure 4 shows comparison of FAGAPSO with the rest of the algorithms in terms of sum throughput in the network for different values of N and M set to 10. The results show that FAGA achieves the highest sum throughput compared to the rest of algorithms. This is because of the better spectrum allocation by FAGA that minimizes interference in the network. According to Shannon channel capacity theorem, reduction in interference improves throughput. As the number of SUs increase, the sum throughput increases for all the algorithms under consideration up to N=800. There is no significant difference between throughput values for N=800 and N=1000. This is because the effect of interference term in the Shannon channel capacity formula (equation (2)) starts becoming significant such that throughput can no longer increase.
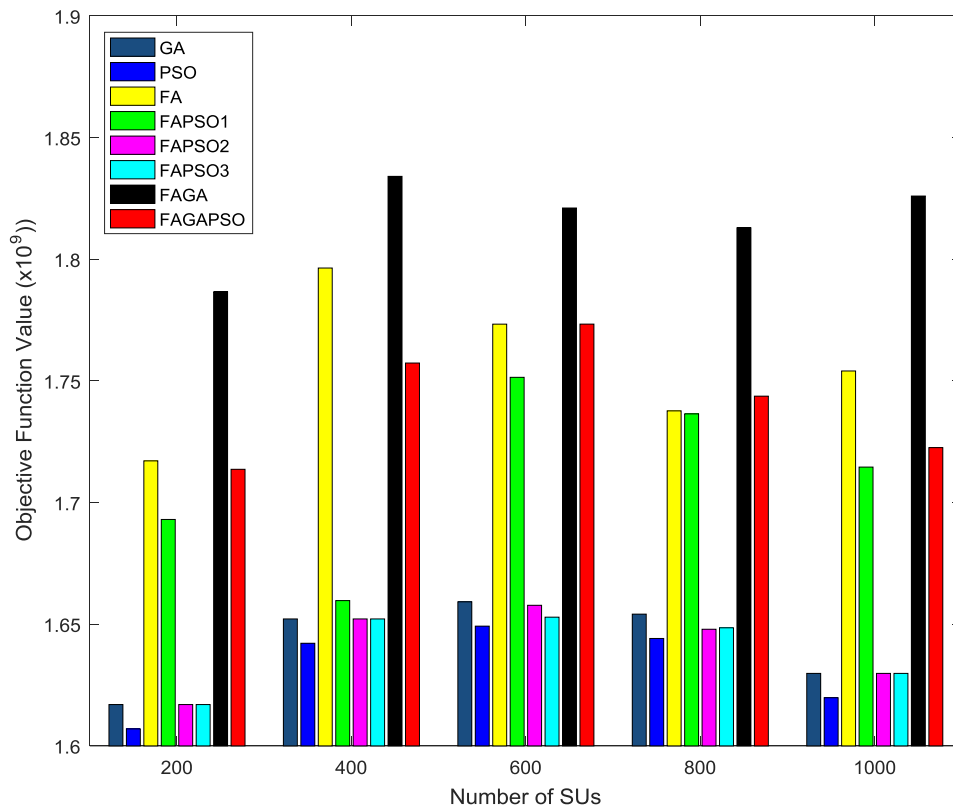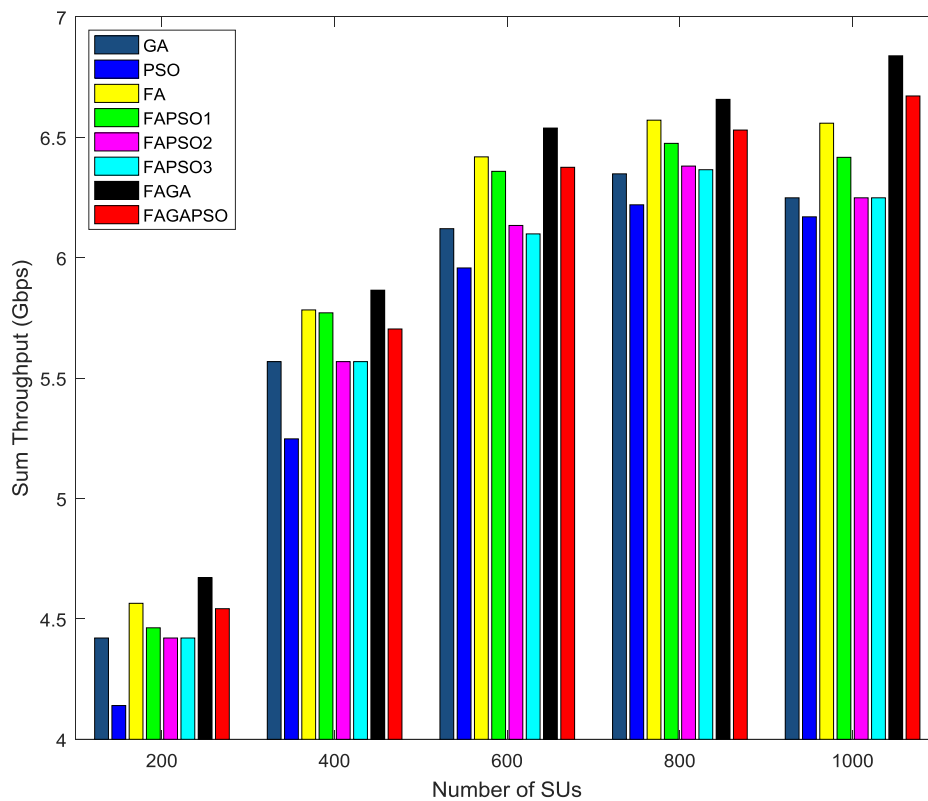
Figure 3. Comparison of Objective Function Values

Figure 4. Comparison of Sum Throughput

5.1.3 Analysis of Performance of Hybrid FA Algorithms for Spectrum Allocation as Binary Optimization Problem

Comparison of FA, GA and PSO: FA outperforms PSO and GA in terms of objective function value and sum throughput for a binary optimization problem. GA outperforms PSO. Firefly movement in FA works better for a binary optimization problem compared to the use P_{best} and g_{best} in PSO as well as mutation and crossover of GA. The results are in agreement with the findings of Liu et al. in [7]) and Anumandla et al. in [6]

Comparison of FAGA with FA and GA: FAGA outperforms FA and GA in terms of objective function value and sum throughput. Results agree with findings by Rahmani A. and Mirhassani S.A. [17] and Luthra J. and Pal Saibal K. [18] that the use crossover feature in FA makes it be able to search the solution space better. This is because, in the FAGA algorithm, the best two algorithms are crossed over before firefly movement so that new chromosomes are generated. The new chromosomes are potentially better solutions that can replace the top four fireflies. At every iteration, if the four new chromosomes have better solutions, they will replace the top four fireflies. The use of GA's crossover feature enables FA to have better exploration ability.

Comparison of FAPSO1 with FA and PSO: FAPSO1 (FA with initial solution of PSO) performs better in terms of objective function value and sum throughput compared to PSO but FA has better performance compared to FAPSO1. Results do not agree with findings by Arunachalam et al[15] that the use of initial solution generated by PSO in FA enables it to generate a better solution. This is because the problem considered by Arunachalam et al[15] was continuous optimization problem. FA is able to generate better solution over 100 iterations than when 50 iterations are used for FA and 50 for PSO. This is because FA outperforms PSO for a binary optimization problem as discussed before. FA has better exploration and exploitation ability compared to PSO for a binary optimization problem.

Comparison of FAPSO2 and FA: FA performs better FAPSO2 (FA with PSO operators) terms of better objective function value and sum throughput. Results do not agree with findings by Kora and Rama [16] that the use of PSO operators in FA during firefly enables it to generate a better solution. This is because the problem considered by Arunachalam et al[15] was continuous optimization problem. FA is able to generate better solutions with its normal firefly movement than with firefly movement towards P_{best} and g_{best}. This implies that for a binary optimization problem, the use of PSO operators during firefly movement in FA degrades the performance of FA. Pure FA has better exploration and exploitation capability compared to the use of PSO operators during firefly movement.

Comparison of FAPSO3 with FA, FAPSO1 and FAPSO2: FAPSO3 matches the performance of FAPSO2 but FAPSO1 outperforms both FAPSO2 and FAPSO3. Compared to FAPSO3, FA outperforms FAPSO3. This implies that the use of PSO operators degrades the performance of FA. As discussed before, FA outperforms PSO for a binary optimization problem. The use initial solution generated by PSO degrades the performance of FA. This is because FA is able to generate better solution over more (100 for the study) iterations than when 50 iterations are used for both FA and PSO. The use of PSO operators during firefly movement also further degrades the performance of FAPSO3.

Comparison of FAGAPSO with All other Algorithms: FAGAPSO matches the performance of FA and FAPSO1 but outperforms FAPSO2 and FAPSO3. This can be explained as follows. Although the use of PSO operators and initial solution in FA degrades its performance, its performance is improved by crossover feature of GA. FAGA outperforms FAGAPSO. This is because the use of GA's crossover feature only in FA is more effective for a binary optimization compared to the use of initial solution of PSO and the use of PSO's P_{best} and g_{best} during firefly movement. This can be attributed to the structure of the spectrum allocation matrix. In the spectrum allocation matrix, only one position has a value of 1 in the spectrum allocation vector of an SU.

Summary: FAGA is superior to all other algorithms including FAGAPSO for a binary optimization problem even when N and M is varied. It can be concluded that for a binary optimization problem, FA has better exploitation and exploration ability when it hybridized with GA compared to when it is hybridized with GA and PSO. Hybridizing FA with PSO degrades it performance for a binary optimization problem.

5.2 Power Allocation

In this section, simulation results for optimization of spectrum allocation using a variety of hybrid FA, PSO and GA algorithms are presented. FAGAPSO is compared with:

- FA
- GA
- PSO
- FAPSO1: FA with initial solution of PSO
- FAPSO2: FA with PSO operators i.e firefly movement using P_{best}, g_{best}, c_1 and c_2 as expressed in equation 22
- FAPSO3: FA with PSO operators with initial solution of PSO
- FAGA: Firefly algorithm with the use crossover feature of GA according to Step 3.3 of Algorithm 2.

100 iterations were used for GA, FA, PSO, FAGA and FAPSO2. In FAPSO1, FAPSO3 and FAGAPSO, 50 iterations were used for both FA and PSO so that the total number of iterations will be also 100. Simulation results are generated for 10 simulation runs and an average is done. The performance of the algorithms is compared using the following metrics: objective function value and sum throughput.

5.2.1 Sum Power

Figure 5 shows performance comparison of FAGAPSO with the rest of the algorithms in terms of sum power in the network for different number of SUs, N and with M set to 10. The figure shows that FAGAPSO achieves the lowest sum power for all the values of N under consideration. The algorithm also achieves lower sum power compared to FAGA,

FAPSO1, FAPSO2 and FAPSO3. This is because FAGAPSO is able to achieve the most optimal power allocation to SUs that minimizes sum power to all SUs according to equation (24). The sum power increases for all the algorithms as N increases because of more active SUs in the network.

5.2.2 Sum Throughput

Figure 6 shows comparison of FAGAPSO with the rest of the algorithms in terms of sum throughput in the network, for different values of N with M set to 10. The figure shows that FAGAPSO achieves the highest sum throughput compared to all the other algorithms. This is because of the improved power allocation that minimizes interference in the network. According to Shannon channel capacity theorem, reduction in interference improves throughput. As N is increased from 200 to 800, there is a steady increase in sum throughput because the effect of interference is not significant. There is no significant difference in throughput as N is varied from 800 to 1000 because the effect of interference term in the Shannon channel capacity formula

becomes more significant such that throughput can longer increase.

5.2.3 Analysis of Performance of Hybrid FA Algorithms for Power Allocation as Continuous Optimization Problem

FAGA has poor performance compared to the FAPSO1, FAPSO2 and FAPSO3. It can be seen from the results PSO is able to search the solution space for a continuous optimization problem better than both GA and FA. This makes a hybrid FA and PSO perform better than hybrid of FA and GA.

FAPSO2 performs better than FAPSO1. This implies that the use of PSO operators in FA allows hybrid FA get a better solution compared to using initial solution generated in PSO in FA. FAPSO3 performs better than FA, GA, PSO, FAPSO1 and FAPSO2 in all the performance metrics. This is because a hybrid FA with the use of initial solution of PSO as well the use of PSO operators (P_{best}, g_{best}, c_1 and c_2 as expressed in equation 3) enables FA to search the solution space better compared with FA, GA, PSO, FAPSO1 and FAPSO2.
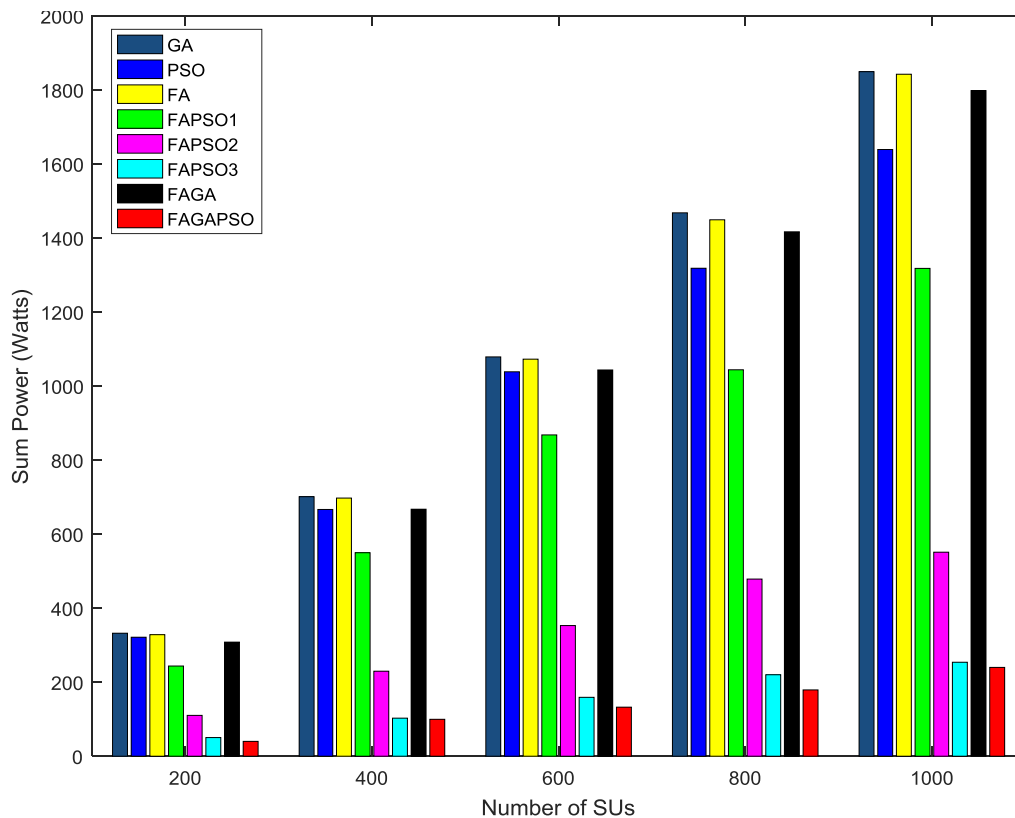

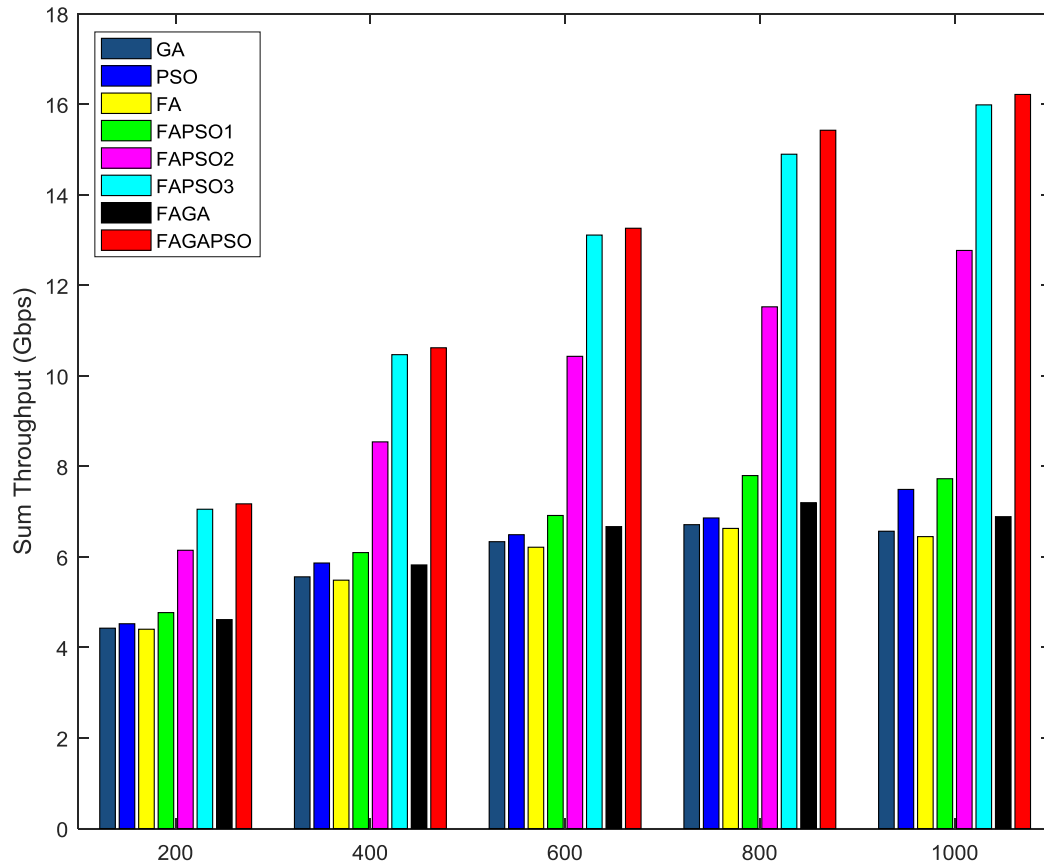*Figure 5.* Comparison of Sum Power

Figure 6. Comparison of Sum Throughput

Compared to other algorithms, FAGAPSO improves all the performance metrics of power allocation except percentage of SUs with SINR less than required threshold where it closely matches that of FAPSO3. This is because the use of crossover feature of GA to mix top ranked fireflies in addition to the use of initial solution of PSO as well as PSO operators in FA further allows it to search the solution space better. This enables FAGAPSO to generate highest sum throughput as well as the lowest sum power and objective function value.

5.3 Analysis and Comparison of Hybrid Firefly Algorithm for Binary and Continuous Optimization

The use of initial solution generated PSO in FA as well as PSO operators in FA is able to improve the final solution power allocation problem but not for spectrum allocation. This can be explained as follows.

As discussed in the previous section, FA is superior compared to PSO for a binary optimization problem. Performance of FA degrades when initial solution of PSO is used in FA because iterations are shared between FA and PSO. This is because FA's firefly movement is more effective for a binary optimization problem than a continuous optimization problem. However, for a continuous optimization problem, the use of initial solution generated by PSO in FA improves FA's performance because PSO performs better compared to FA for a continuous optimization problem.

The use of PSO's P_{best} and g_{best} in FA during firefly movement degrades its performance for a binary optimization problem. FA is able to search the solution space better for a binary optimization problem with its normal firefly movement than with the use of PSO's P_{best} and g_{best} during firefly movement.

FAGAPSO outperforms all other algorithms under consideration for power allocation but for spectrum allocation FAGA outperforms all other algorithms. This is because for power

allocation the use of GA's crossover feature in addition to the use of PSO's initial solution in FA as well as the use of PSO's P_{best} and g_{best} during firefly movement is able to generate a better solution compared to FA only for a continuous optimization problem through better exploration and exploitation ability. However, for a binary optimization problem, the use of GA's crossover feature only in FA, improves its performance compared to the use PSO's initial solution in FA as well as PSO's P_{best} and g_{best} during firefly movement. This is mainly because of the structure of the spectrum allocation matrix whereby only one position has a value of 1 in the spectrum allocation vector of an SU.

6. Conclusion

The results have shown that FAGAPSO outperforms all other algorithms for power allocation as a binary optimization problem. Results also show that for spectrum allocation as a binary optimization problem, FAGA is superior to FAGAPSO, FA, GA, PSO and the three other hybrid FA algorithms. These results demonstrate that the performance of hybrid population based metaheuristic algorithms depend on whether the optimization problem is binary or continuous.

References:

[1] G. J. Woeginger, "Exact Algorithms for NP-Hard Problems: A Survey," in *Combinatorial Optimization — Eureka, You Shrink!*, vol. 2570, M. Jünger, G. Reinelt, and G. Rinaldi, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2003, pp. 185–207.

[2] Z. Beheshti and S. M. H. Shamsuddin, "A Review of Population-based Meta-Heuristic Algorithm," *International Journal of Advances in Soft Computing Applications*, vol. 5, no. 1, p. 36, 2013.

[3] S. K. Goudos, "Evolutionary Algorithms for Wireless Communications — A Review of the State-of-the art," in *Contemporary Issues in Wireless Communications*, M. Khatib, Ed. InTech, 2014.

[4] P. J. Gaidhane and M. J. Nigam, "A hybrid grey wolf optimizer and artificial bee colony algorithm for enhancing the performance of complex systems," *Journal of Computational Science*, vol. 27(1), pp. 284–302, Jul. 2018, doi: 10.1016/j.jocs.2018.06.008.

[5] N. Singh and S. B. Singh, "Hybrid Algorithm of Particle Swarm Optimization and Grey Wolf Optimizer for Improving Convergence Performance," *Journal of Applied Mathematics*, vol. 2017(1), pp. 1–15, 2017, doi: 10.1155/2017/2030489.

[6] K. K. Anumandla, S. Kudikala, B. A. Venkata, and S. L. Sabat, "Spectrum allocation in cognitive radio networks using firefly algorithm," in *International Conference on Swarm, Evolutionary, and Memetic Computing*, 2013, pp. 366–376, Accessed: Oct. 24, 2016. [Online]. Available: http://link.springer.com/chapter/10.1007/978-3-319-03753-0_33.

[7] Q. Liu, W. Lu, and W. Xu, "Spectrum Allocation Optimization for Cognitive Radio Networks Using Binary Firefly Algorithm," Quebec, Canada, Aug. 2014.

[8] Z. Zhao, Z. Peng, S. Zheng, and J. Shang, "Cognitive radio spectrum allocation using evolutionary algorithms," *IEEE Transactions on Wireless Communications*, vol. 8(9), no. 9, pp. 4421–4425, Sep. 2009, doi: 10.1109/TWC.2009.080939.

[9] J. Elhachmi and Z. Guennoun, "Cognitive radio spectrum allocation using genetic algorithm," *EURASIP Journal on Wireless Communications and Networking*, vol. 2016(1), no. 1, Dec. 2016, doi: 10.1186/s13638-016-0620-6.

[10] Z. Jie and L. Tiejun, "Spectrum Allocation in Cognitive Radio with Particle Swarm

Optimization Algorithm," *Chinese Scientific Papers Online*, 2012, Accessed: Aug. 18, 2015. [Online]. Available: http://www.paper.edu.cn/en_releasepaper/downPaper/201201-658.

[11] F. Koroupi, S. Talebi, and H. Salehinejad, "Cognitive radio networks spectrum allocation: An ACS perspective," *Scientia Iranica*, vol. 19(3), no. 3, pp. 767–773, Jun. 2012, doi: 10.1016/j.scient.2011.04.029.

[12] K. Ronoh, G. Kamucha, T. Olwal, and T. Omwansa, "Improved Resource Allocation for TV White Space Network Based on Modified Firefly Algorithm," *Journal of Computing and Information Technology*, vol. 26(3), no. 3, pp. 167–167, Sep. 2018.

[13] R. Kennedy, K. George, O.-O. William, O. Thomas, and O. Tonny, "Firefly algorithm based power control in wireless TV white space network," in *AFRICON, 2017 IEEE*, 2017, pp. 155–160.

[14] K. Ronoh, G. Kamucha, and T. Omwansa, "Comparison of Hybrid Firefly Algorithms for Power Allocation in a TV White Space Network," *IJCA*, vol. 178, no. 38, pp. 37–43, Aug. 2019, doi: 10.5120/ijca2019919264.

[15] S. Arunachalam, T. AgnesBhomila, and M. Ramesh Babu, "Hybrid Particle Swarm Optimization Algorithm and Firefly Algorithm Based Combined Economic and Emission Dispatch Including Valve Point Effect," in *Swarm, Evolutionary, and Memetic Computing*, vol. 8947, B. K. Panigrahi, P. N. Suganthan, and S. Das, Eds. Cham: Springer International Publishing, 2015, pp. 647–660.

[16] P. Kora and K. S. Rama Krishna, "Hybrid firefly and Particle Swarm Optimization algorithm for the detection of Bundle Branch Block," *International Journal of the Cardiovascular Academy*, vol. 2(1),

no. 1, pp. 44–48, Mar. 2016, doi: 10.1016/j.ijcac.2015.12.001.

[17] I. Fister, I. Fister, X.-S. Yang, and J. Brest, "A comprehensive review of firefly algorithms," *Swarm and Evolutionary Computation*, vol. 13(1), pp. 34–46, Dec. 2013, doi: 10.1016/j.swevo.2013.06.001.

[18] J. Luthra and S. K. Pal, "A hybrid Firefly Algorithm using genetic operators for the cryptanalysis of a monoalphabetic substitution cipher," in *2011 World Congress on Information and Communication Technologies*, Mumbai, India, Dec. 2011, pp. 202–206, doi: 10.1109/WICT.2011.6141244.

[19] R. Kennedy, O. Tonny, and K. George, "Novel Resource Allocation Algorithm for TV White Space Networks Using Hybrid Firefly Algorithm," *IJC*, vol. 32, no. 1, p. 20, 2019.

[20] B. V. Kumar, "Bat Algorithm and Firefly Algorithm for Improving Dynamic Stability of Power Systems Using UPFC," *International Journal on Electrical Engineering and Informatics*, vol. 8(1), no. 1, pp. 164–188, Mar. 2016, doi: 10.15676/ijeei.2016.8.1.12.

[21] M. Elkhechafi, H. Hachimi, and Y. Elkettani, "A new hybrid cuckoo search and firefly optimization," *Monte Carlo Methods and Applications*, vol. 24(1), no. 1, pp. 71–77, Mar. 2018, doi: 10.1515/mcma-2018-0003.

[22] A. Layeb and Z. Benayad, "A Novel Firefly Algorithm Based Ant Colony Optimization for Solving Combinatorial Optimization Problems," *International Journal of Computer Science and Applications*, vol. 11(2), no. 2, p. 19, 2014.

[23] X.-S. Yang, "Firefly algorithms for multimodal optimization," in *International Symposium on Stochastic Algorithms*, 2009, pp. 169–178, Accessed: Nov. 21, 2016. [Online]. Available:

http://link.springer.com/chapter/10.1007/9
78-3-642-04944-6_14.

[24] S. Arora and S. Singh, "A conceptual comparison of firefly algorithm, bat algorithm and cuckoo search," in *2013 International Conference on Control, Computing, Communication and Materials (ICCCCM)*, Allahabad, India, Aug. 2013, pp. 1–4, doi: 10.1109/ICCCCM.2013.6648902.

[25] E. Elbeltagi, T. Hegazy, and D. Grierson, "Comparison among five evolutionary-based optimization algorithms," *Advanced Engineering Informatics*, vol. 19(1), no. 1, pp. 43–53, Jan. 2005, doi: 10.1016/j.aei.2005.01.004.

[26] D. Gurney, G. Buchwald, L. Ecklund, S. Kuffner, and J. Grosspietsch, "Geo-location database techniques for incumbent protection in the TV white space," in *3rd IEEE Symposium on New Frontiers in Dynamic Spectrum Access Networks, 2008. DySPAN 2008.*, 2008, pp. 1–9, Accessed: Sep. 05, 2015. [Online]. Available: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=4658242.

[27] L. A. Vardhan and A. Vasan, "Evaluation of penalty function methods for constrained optimization using particle swarm optimization," in *2013 IEEE Second International Conference on Image Information Processing (ICIIP)*, 2013, pp. 487–492, Accessed: Mar. 13, 2017. [Online]. Available: http://ieeexplore.ieee.org/abstract/document/6707639/.

Contribution of individual authors to the creation of a scientific article (ghostwriting policy)

Kennedy Ronoh carried out the programming and simulation and also wrote the original draft.

George Kamucha was responsible for the draft review, supervision and validation.

Creative Commons Attribution License 4.0 (Attribution 4.0 International, CC BY 4.0)