Innovative Power Allocation Strategy for NOMA Systems by Employing the Modified ABC Algorithm

Sunkaraboina SREENU, Naidu KALPANA
Dept. of ECE, National Institute of Technology, Warangal, India
sreenus@student.nitw.ac.in, kalpana@nitw.ac.in
Submitted February 10, 2022 / Accepted June 20, 2022 / Online first July 15, 2022

Abstract. Non-Orthogonal Multiple Access (NOMA) technique is a remarkable component of 5G wireless networks; since NOMA immensely augments the spectral efficiency and serves all users fairly. To accomplish these, efficient power allocation is crucial for improving the NOMA system’s performance. Accordingly, in this article, we formulate a power allocation optimization issue, which concentrates on enriching the system sum-throughput, by realizing the transmitted power constraint and also fulfilling the minimum throughput for each user. However, to tackle this mentioned optimization problem, a Modified Artificial Bee Colony (MABC) algorithm is proposed. Besides, the designed MABC algorithm obtains optimal powers among multiplexed users on every sub-channel. Further, simulation results illustrate that the presented power allocation scheme-based NOMA system’s sum throughput is higher than the original ABC-based power allocation and other state-of-the-art power allocation schemes. Moreover, the MABC method swiftly converges to optimal solutions compared to the original ABC algorithm under selected control parameters.

Keywords
NOMA, artificial bee colony, power allocation, sum rate

1. Introduction
The ongoing enormous growth in mobile traffic and ubiquity of smart devices have placed higher demands on the upcoming (5G) wireless systems [1], [2]. In order to meet these emerging cellular industry challenges, the 5G networks need to enhance system capacity further. To this end, Non-Orthogonal Multiple Access (NOMA) has evolved as one of the essential technologies for 5G systems due to its high bandwidth utilization, user fairness, and massive user connectivity [3–6]. Further, contrary to the traditional Orthogonal Multiple Access (OMA) [7–13]; NOMA breaks the orthogonality between users by exploiting the power domain multiplexing, and it serves multiple users simultaneously at different power levels [14].

Moreover, in a downlink NOMA framework, the Base Station (BS) superimposes various users’ data for simultaneous transmission over the single radio resource unit [15]. As a result, superimposed data for multiple users can be separated at the receivers (of the corresponding users) by using the Successive Interference Cancellation (SIC) method. However, to ensure successful SIC detection at each user’s receiver, users having poor channel gains must be assigned with higher powers than those powers allotted for good channel gain users [16], [17]. Furthermore, the NOMA system can control the user multiplexing and transmission throughput by allocating distinct powers to different users. Therefore, user power allocation becomes a significant issue in addition to user scheduling problems [18], [19].

In recent literature, numerous Power Allocation (PA) techniques for NOMA networks have been investigated to heighten the sum capacity of the system. In [20], [21]; Fixed Power Allocation (FPA) scheme has been exploited for power optimization to enhance system capacity. But, due to fixed power factors between the users, FPA does not provide an optimal solution. In consequence, Fractional Transmit Power Allotment (FTPAl) was investigated in [22–25], which is a widely used dynamic power allocation algorithm. Accordingly, FTPA allocates powers to users as per the channel conditions (of users). However, in order to obtain the best performance, FTPA requires prior computer simulations for determining the decay factor of channel gains. Analogously, [26] presents the difference-of-convex programming-based inter and intra-sub-channel power allocation for optimizing the system’s sum capacity. In any case, the proposed approach in [26] delivers a sub-optimal solution, even though it outperforms OMA and FTPA techniques.

Additionally, to achieve user fairness, the fair power allocation method was proposed in [27], [28]; that delivers the capacity to every user exceeding the corresponding OMA capacity by ensuring the overall power constraint. Analogously, in [29], authors explored two power allotment methods to optimize the ergodic capacity in two-user NOMA networks with total transmission power and weak user’s minimum data throughput constraints. However, in [29], the proposed method did not guarantee the strong user’s minimal data throughput. Further, in [30], Karush-Kuhn-Tucker
optimal conditions have been employed to derive the closed-form results of power allocation factors among multiplexed users on each sub-channel.

However, most of the works discussed above accomplish significant results by either relaxing power allocation constraints or offering a sub-optimal solution. In contrast, Evolutionary Algorithms (EA) can prove to be good assets in handling complex non-linear constrained optimisation problems because they repeatedly renew an initial solution up to the result becomes an optimal solution. Nevertheless, only a very few studies [31–33] have exploited EAs namely: particle swarm optimization, salp swarm optimization, and genetic algorithm to improve the performance of NOMA systems. Hitherto, no study has reported utilizing the Artificial Bee Colony (ABC) algorithm to the issues related to power allocation in the NOMA systems. But, the motivation for adopting ABC (to the NOMA system) is the fact that it requires limited control parameters, and it provides the best solution compared to the other meta-heuristic algorithms [34], [35]. Nonetheless, when dealing with complex problems, ABC algorithm confronts unacceptable exploitation [36]. To overcome this drawback, we developed a Modified ABC (MABC) method and utilized it to enhance the NOMA system performance as given in the below paragraph.

In this article, we propose unique solution search equations by incorporating the best solution in the “employed and onlooker bees phases”, respectively, to enhance exploitation and retain exploration. After that, the MABC approach is operated to address the PA problem, thereby increasing the total throughput of the downlink NOMA system. Ultimately, simulation findings validate that the projected method for NOMA system outperforms the original ABC-based PA, FTPA algorithms, and OMA system. Moreover, MABC shows fast convergence compared to the original ABC method.

The remainder of this article is organized as follows. Section 2 provides the NOMA conceptual framework and formulation of the power allotment problem. Then, the proposed MABC-based power allocation strategy is discussed in Sec. 3. Subsequently, Section 4 validated the proposed MABC with numerical results based on standard test functions and also provides simulations of MABC-based power allocation for NOMA system. Eventually, in Sec. 5, the article is concluded.

2. Downlink System Design and Problem Formulation

To provide explicit elucidation, this section is partitioned into two sub-sections. Section 2.1 gives a brief overview of the downlink NOMA transmission paradigm. Following that, the power allocation optimization problem (to improve the sum-rate) is mathematically developed in Sec. 2.2.
Consider that BS has precisely known Channel State Information (CSI). Consequently, according to this identified CSI, BS delivers the multiplexed coded symbols to all users. For each sub-channel; $s_{w,m}$ and $s_{s,m}$ represent the modulation symbols of $UE_w$ and $UE_s$, respectively and the multiplexed symbol for the sub-channel $m$ is given as

$$x_m = \sqrt{\alpha_{w,m} P_m} s_{w,m} + \sqrt{\alpha_{s,m} P_m} s_{s,m}$$  \hspace{1cm} (1)

where $\alpha_{w,m} & \alpha_{s,m}$ represent the power allocation factors of $UE_w$ and $UE_s$ in sub-channel $m$ and $\alpha_{w,m} + \alpha_{s,m} = 1$. Further, $P_m$ is the power for sub-channel $m$, and is identical over all the sub-channels (i.e., $P_m = \frac{P}{M}$). The superposed signal received at arbitrary user $n$ is written as

$$y_{n,m} = h_{n,m} x_m + v_{n,m}.$$  \hspace{1cm} (2)

In the above equation, $h_{n,m} = g_{n,m} d_{n}^{-\gamma}$ gives the complex channel gain in between the BS and $n^{th}$ UE on sub-channel $m$. Further, Rayleigh fading coefficient is indicated with $g_{n,m}$. Besides, $d_{n}$ is the distance from BS to UE $n$. Moreover, $\gamma$ gives path loss slope, and $v_{n,m}$ constitutes Gaussian noise. With no loss of generality, all user’s channel qualities are organized as follows: $|h_1|^2 \leq |h_2|^2 \leq ... \leq |h_N|^2$.

Further, the main idea of NOMA is to allocate high power to $UE_w$ and less power to $UE_s$. For sub-channel $m$, if $|h_{w,m}|^2 \leq |h_{s,m}|^2$ then corresponding power allocation factors are $\alpha_{w,m} > \alpha_{s,m}$. Moreover, weak user signal is directly decodable due to high power allocation. Thus, $s_{w,m}$ is first decoded and removed from superimposed received signal, then $UE_s$ decodes $s_{s,m}$. Therefore, $UE_s$ extracts its signal without multi user distortion with the assumption of perfect SIC. However, $UE_w$ is not able to decode $s_{s,m}$ and remove it from multiplexed signal due to low power assigned to $UE_s$. For this reason, $UE_w$ treats $s_{s,m}$ as noise and detects its own signal $s_{w,m}$. In other words, $UE_w$ does not perform SIC. Hence, users’ data rates in $m^{th}$ sub-channel are represented as

$$R_{w,m} = B \log_2 \left( 1 + \frac{\alpha_{w,m} P_m |h_{w,m}|^2}{N_0 + \alpha_{s,m} P_m |h_{s,m}|^2} \right),$$  \hspace{1cm} (3)

$$R_{s,m} = B \log_2 \left( 1 + \frac{\alpha_{s,m} P_m |h_{s,m}|^2}{N_0} \right).$$  \hspace{1cm} (4)

In (3) and (4), $N_0$ is noise power and it is given by $N_0 = kTB_{sc}$. Here, $k, T$ are Boltzmann’s constant, temperature in degree Kelvin respectively. Hence, the achievable sum rate is obtained by summing all the sub-channel’s total rate, which is given as

$$R_{\text{sum}} = \sum_{m=1}^{M} (R_{s,m} + R_{w,m}).$$  \hspace{1cm} (5)

### 2.2 Problem Formulation

The ultimate goal of power assignment problem is to enhance the system’s sum throughput while preserving the total transmit power and each user’s required rate constraints [38], [39]. Therefore, the sum rate optimization problem is formulated as:

$$\text{Max} \quad R_{\text{sum}},$$  \hspace{1cm} (6)

subject to: $\alpha_{s,m} + \alpha_{w,m} = 1,$  \hspace{1cm} (6a)

$\alpha_{w,m} > \alpha_{s,m},$  \hspace{1cm} (6b)

$R_{s,m} \geq R_{s,m}^{\text{min}},$  \hspace{1cm} (6c)

$R_{w,m} \geq R_{w,m}^{\text{min}},$  \hspace{1cm} (6d)

and $0 \leq \alpha_{w,m}, \alpha_{s,m} \leq 1$  \hspace{1cm} (6e)

where (6a) represents the sum of the $m^{th}$ sub-channel user’s powers which is equal to $P_m$ (i.e., $\alpha_{s,m} P_m + \alpha_{w,m} P_m = P_m$). Besides, (6b) indicates the basic NOMA principle that more power is allocated $UE_w$ than $UE_s$. Moreover, (6c) and (6d) represents minimum data rate requirement of $UE_s$ and $UE_w$ respectively. Subsequently, (6e) gives bounds of the power allocation factors of each sub-channel user.

### 3. Standard ABC Algorithm and its Modified Version for Optimal Power Allocation

#### 3.1 Standard ABC Algorithm

The ABC algorithm is a swarm-pertained optimization scheme that resembles the intelligent foraging demeanor of honey bees and has been used extensively for several practical problems. It was developed by Karaboga as part of the swarm intelligence algorithms family [40]. In the ABC, three kinds of honey bees are involved in the optimization process: employed, onlooker (observer), and scout bees. First, the employed bees search for food near the food source preserved in their memory; in the meantime, they communicate this knowledge about these food sources with the observer bees. Then, observer bees calculate fitness (nectar amount) and choose the best food sources discovered by employed bees. After that, the food source which has not improved the fitness is abandoned by scout bees and replaced that solution with a random food source to enrich the exploration.

The standard ABC algorithm solves any optimization problem in four phases:

1. **Initialization**
2. **Employed Bees**
3. **Onlooker Bees**
4. **Scout Bees**

The steps required for all the phases are outlined below:
1. **Initialization:**

In this phase, ABC produces a random number of NP food sources (solutions). Here, NP indicates size of bees, which is same for both employed and onlooker bees. Let’s take the \( j \)th food source as \( \alpha_i = \{\alpha_{i1}, \alpha_{i2}, \ldots, \alpha_{iD}\} \), where \( D \) is the optimization variables’ dimension. Each solution is obtained within the range of \( j \)th index by

\[
\alpha_{i,j} = \alpha_{j}^{\text{min}} + \psi(\alpha_{j}^{\text{max}} - \alpha_{j}^{\text{min}}).
\]

Here, \( \alpha_{j}^{\text{min}} \) and \( \alpha_{j}^{\text{max}} \) denote minimum and maximum limits for index \( j \), respectively. Besides, \( \psi \) represents randomly generated value which lies in the range \([0, 1]\), \( i = \{1, 2, \ldots, NP\} \), and \( j = \{1, 2, 3, \ldots, D\} \). Then every solution is evaluated by

\[
\text{fit}_i = \begin{cases} 
1 & f_i > 0 \\
1 + |f_i|, & f_i < 0
\end{cases}
\]

where \( \text{fit}_i, f_i \) indicates the fitness and objective functions of food source \( i \) respectively. The cost function can be utilized directly as a fitness function for maximizing problems.

2. **Employed Bees:**

Employed bees produce the neighborhood solution for every initial solution. The position of new solution can be computed as

\[
\beta_{i,j} = \alpha_{i,j} + \lambda_{i,j}(\alpha_{i,j} - \alpha_{k,j}).
\]

In (9), \( j \) is a random integer and \( j \in \{1, 2, 3, \ldots, D\} \); \( k \) is selected partner solution & \( k \in \{1, 2, \ldots, NP\}, k \neq i \). \( \lambda_{i,j} \) represents randomly generated parameter, which is lies between \(-1 \) and \( 1 \). Note that \( \beta_i \) is different from \( \alpha_i \) only at the \( j \)th component. Then, fitness value of the updated solution is evaluated, and compares with the previous solution \( \alpha_i \). If the updated fitness is greater as compared with old solution fitness, \( \alpha_i \) is replaced with the \( \beta_i \). Otherwise, the previous solution would be carried on.

3. **Onlooker Bees:**

After employed bees have completed their quests, they communicate knowledge about fitness and food source locations with onlooker bees. Then, each onlooker bee chooses a solution according to the probability \( P_l \) and it is determined by

\[
P_l = \frac{\text{fit}_i}{\sum_{i=1}^{NP} \text{fit}_i}.
\]

Once the onlookers select the food source position, it produces a new solution and corresponding fitness by utilizing (9) and (8) respectively. Subsequently, the greedy mechanism is performed between new and old food sources similar to the employed bees.

4. **Scout Bees:**

If a solution has not been updated within a given number of trials, it is exited, and the related bee evolves into a scout bee. Then, it generates a random solution by using (7). Finally, evaluate the better solution \( \alpha_{\text{best}} \) and objective function \( f_{\text{best}} \) and repeat the phases: (2)–(4) until the total number of iterations. For easier comprehension, the ABC flow diagram is depicted in Fig. 2.

### 3.2 Proposed Modified ABC Algorithm

As previously stated, the ABC algorithm faces some difficulties in balancing exploration and exploitation and also the slow convergence when solving complex constrained optimization problems. To overcome the slow convergence rate and improve the performance, the current position of “employed and onlooker bees” is updated by incorporating the best solution position in the modified ABC algorithm. Hence, MABC produces the new solution nearer to the best solution in both bee phases (employed and onlooker) for every iteration. Therefore, the modified solution search equation for the “employed and onlooker bees phase” is represented as

\[
\beta_{i,j} = \alpha_{\text{best},j} + \lambda_{i,j}(\alpha_{i,j} - \alpha_{k,j}).
\]

In (11), \( \alpha_{\text{best}} \) is the best solution before finding the new solution of each phase (both employed and onlookers bees). Moreover, food sources and the objective function of the algorithm represent the power allocation factors and sum throughput, respectively.
After all the users are paired to each sub-channel according to user pairing scheme. The propounded MABC-based Power Allocation (MABC-PA) assigns optimized powers to each paired user of the sub-channel to maximize sum throughput. The steps required for MABC-PA method for downlink NOMA system are detailed below.

Step 1: Set the initial parameters of the MABC-PA scheme such as paired users channel gain values of \( m \)th sub-channel, BS total power (\( P_t \)), number of users (\( N \)), system bandwidth (\( B \)), population size (\( NP \)), dimension of NP (\( D \)), number iterations (\( T \)) and limit.

Step 2: Randomly generate the population matrix with in the bounds (i.e power allocation factors matrix) using \( \alpha_{i,j} = a_{\text{min}} + \psi (a_{\text{max}} - a_{\text{min}}) \), \( i \in NP \) and \( j \in D \).

Step 3: Evaluate the objective function of each solution by using \( R_{m,i} = (R_{s,j} + R_{c,i}) \). If any solution do not satisfy the constraints (mentioned in (6a)–(6d)) then the penalty is added to the corresponding objective function [41].

Step 4: Memorize the best objective function and best solution.

Step 5: ‘Employed Bees’ update the every solution by using modified solution search equation \( \beta_{i,j} = a_{\text{best},j} + \lambda_{i,j}(\alpha_{i,j} - \alpha_{k,j}) \).

Step 6: Find the objective function for the updated solution and compare with the older solution. If the updated solution sum rate (objective function) is better than older solution. Then replace older solution with updated solution and memorize the best objective function and solution.

Step 7: ‘Onlooker Bees’ select solution based on the probability (\( P_i \)), which can be obtained by using formula \( P_i = \frac{R_{m,i}}{\sum R_{m,i}} \).

Step 8: Once the onlooker bee identifies the solution position, it produces the new solution similar to the ‘Employed Bees’ Then, find the objective function (\( R_{m,i} \)). Subsequently, perform greedy selection procedure and then replace the older solution with the better solution.

Step 9: If any solution not updated in Employed and Onlooker Bees Phases for a particular limit. Then, Scout Bees replace that solution with random solution which can be computed by using \( \alpha_{i,j} = a_{\text{min}} + \psi (a_{\text{max}} - a_{\text{min}}) \).

Step 10: Finally, compute the better objective function and corresponding solution. This process (steps: 5 to 9) repeated for given number of iterations.

Step 11: Furthermore, entire process is repeated for all the \( M \) sub channels. Then, system’s total throughput is obtained by combining the sum rate of all the sub-channels (i.e., \( R_{\text{sum}} = \sum_{m=1}^{M} R_m \)).

The summary of the presented power allocation algorithm is described in Algorithm 1 is in accordance with the presented sum-throughput maximization problem.

\[ \text{Algorithm 1. Proposed MABC-based Power Allocation Algorithm} \]

\textbf{Require:} Control Parameters: Population size (\( NP \)), Number of iterations (\( T \)), limit. 
\textbf{System Parameters:} \( N, M, B_{sc}, N_a, P_m, g_{s,m}, \text{and } g_{w,m} \).

\textbf{Ensure:} Optimal power allocation factors and Sum rate.

1: \textbf{for} \( i = 1 : NP \) \textbf{do}
2: \hspace{1em} Generate random solution by using (7);
3: \hspace{1em} Evaluate the objective function by (6);
4: \hspace{1em} \textbf{end for}
5: \textbf{iteration} = 1;
6: \textbf{while} (\text{iteration } \leq T) \textbf{do}
7: \hspace{1em} //Employed Bee Phase:
8: \hspace{2em} \textbf{for} \( i = 1 : NP \) \textbf{do}
9: \hspace{3em} Produce new solution \( \beta_i \) of the solution \( \alpha_i \) for employed bees using (11);
10: \hspace{3em} Compute objective function (sum rate) and fitness function \( \beta_i \);
11: \hspace{3em} Perform greedy selection procedure among \( \beta_i \) and \( \alpha_i \), then retain with best solution;
12: \hspace{3em} If solution is not updated \( \text{trail}_i \leftarrow \text{trail}_i + 1 \),
13: \hspace{3em} else \( \text{trail}_i \leftarrow 0 \);
14: \hspace{2em} \textbf{end for}
15: \hspace{1em} Memorize the best solution;
16: \hspace{1em} Evaluate probability \( P_i \) by (10) for fitness functions of employed bee solutions;
17: \hspace{1em} //Onlooker Bee Phase:
18: \hspace{2em} \( \eta = 0 \); \( i = 1 \);
19: \hspace{2em} \textbf{while} (\( \eta < NP \)) \textbf{do}
20: \hspace{3em} If \( \text{rand} < P_i \) then
21: \hspace{4em} \hspace{1em} Generate new \( \beta_i \) by (11) for onlooker bee;
22: \hspace{4em} \hspace{1em} Perform greedy selection approach between \( \beta_i \) and \( \alpha_i \) and then select best solution;
23: \hspace{4em} \hspace{1em} If solution is not updated \( \text{trail}_i \leftarrow \text{trail}_i + 1 \),
24: \hspace{4em} \hspace{1em} else \( \text{trail}_i \leftarrow 0 \);
25: \hspace{3em} \hspace{1em} \( \eta \leftarrow \eta + 1 \);
26: \hspace{3em} \hspace{1em} \textbf{end if}
27: \hspace{2em} \textbf{end while}
28: \hspace{1em} //Scout Bee Phase:
29: \hspace{2em} If \( \text{max(trail}_i) > \text{limit} \) then
30: \hspace{3em} Update the \( \alpha_i \) with new solution obtained solution by (7);
31: \hspace{3em} \textbf{end if}
32: \hspace{1em} Store better solution obtained so far;
33: \hspace{1em} \textbf{end while}
Besides, Theorem 1 shows that optimal powers are provided by the MABC algorithm.

**Theorem 1** Proposed MABC algorithm induces optimal powers for each sub channel’s users to maximize the system sum throughput.

**Proof.** To solve the sum-rate maximization problem given in (6); MABC is employed in this work. However, in order to optimize the system’s sum throughput, MABC finds the optimal powers for all users in the following way:

- The users are paired to every sub-channel (for receiver simplicity, two users are paired in each sub-channel). For the given control and system parameters, Algorithm 1 produces the appropriate power factors $\alpha_{s,m}$, $\alpha_{u,m}$ for the sub-channel $m$ by satisfying all the constraints.
- Algorithm 1 renews the power factors in every iteration and stores the better result. Hence, it provides optimal power to every user on each sub-channel at the end of the $T$ iterations.

Hence, it is proved.

### 4. Numerical Results and Discussion

#### 4.1 Experimental Validation of Proposed MABC Algorithm

In order to corroborate the MABC algorithm’s performance, we took the eight distinct benchmark test functions consisting of four Uni-Modal (UM) and four Multi-Modal (MM) functions as displayed in Tab. 1. The uni-modal functions are mainly used to analyze the convergence performance of any algorithm. On the other hand, multi-modal functions play a vital role in testing whether the proposed algorithm is escaping from the local optimum solution efficiently. Besides, in Tab. 1, the parameter $D$ denotes the dimensionality of solution search space.

The significance of the proposed MABC method is determined by comparing its results to the classic ABC [40]. In order to make an accurate comparison with the original ABC, both algorithms were studied utilizing the identical control parameters: the size of population $NP = 30$, limit $= NP \times D$, number of iterations $T = 1000$.

To ensure that the experimental results are consistent, we performed each algorithm 30 times and averaged the results. We evaluated the five statistical parameters such as Best, Average, Median, Standard Deviation (SD), and Worst solutions for comparing the performance of ABC and MABC algorithms. Table 2 and Table 3, respectively, illustrate the experimental results produced by each algorithm for $D = 10$ and $D = 30$ dimensions. As a result of the numerical findings, the MABC performs admirably on almost all of the benchmark functions except $f_2$. Since, the second part $100(x_{i+1} - x_i)^2$ in $f_2$ (Rosenbroack function) significantly impacts the value of the function.

Furthermore, the solution accuracy of MABC and the ABC algorithm are identical for functions $f_3$ (Rastrigen) and $f_6$ (Schwefel) with $D = 10$. When the dimension is smaller, the multi-modal functions $f_5$ and $f_6$ easily obtain the optimal solutions. The complexity of achieving the optimal solution gradually increases as the dimension grows larger. However, even with larger dimensions ($D = 30$), MABC provides better solution accuracy for the functions $f_5$, $f_6$. In addition, Figure 3 demonstrates how the average of the Best Fitness Value of both the schemes (i.e., MABC and ABC) varies

<table>
<thead>
<tr>
<th>Function Name</th>
<th>Mathematical Expression</th>
<th>Type</th>
<th>Search Range</th>
<th>$f_{min}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sphere</td>
<td>$f_1(\mathbf{x}) = \sum_{i=1}^{D} x_i^2$</td>
<td>UM</td>
<td>$[-100, 100]^D$</td>
<td>0</td>
</tr>
<tr>
<td>Rosenbrock</td>
<td>$f_2(\mathbf{x}) = \sum_{i=1}^{D} (x_i - 1)^2 + 100(x_{i+1} - x_i)^2$</td>
<td>UM</td>
<td>$[-30, 30]^D$</td>
<td>0</td>
</tr>
<tr>
<td>Sum Squares</td>
<td>$f_3(\mathbf{x}) = \sum_{i=1}^{D} x_i^2$</td>
<td>UM</td>
<td>$[-10, 10]^D$</td>
<td>0</td>
</tr>
<tr>
<td>Step</td>
<td>$f_4(\mathbf{x}) = \sum_{i=1}^{D} x_i^2 + 0.5$</td>
<td>UM</td>
<td>$[-100, 100]^D$</td>
<td>0</td>
</tr>
<tr>
<td>Rastrigin</td>
<td>$f_5(\mathbf{x}) = 100 + \sum_{i=1}^{D} x_i^2 - 10\cos(2\pi x_i)$</td>
<td>MM</td>
<td>$[-5.12, 5.12]^D$</td>
<td>0</td>
</tr>
<tr>
<td>Schwefel</td>
<td>$f_6(\mathbf{x}) = -\sum_{i=1}^{D} \left</td>
<td>x_i \sin \sqrt{</td>
<td>x_i</td>
<td>} \right</td>
</tr>
<tr>
<td>Ackley</td>
<td>$f_7(\mathbf{x}) = e + 20 - 20\exp \left(-0.2 \sqrt{\frac{1}{D} \sum_{i=1}^{D} x_i^2} \right) - \exp \left(\frac{1}{D} \sum_{i=1}^{D} \cos(2\pi x_i) \right)$</td>
<td>MM</td>
<td>$[-32, 32]^D$</td>
<td>0</td>
</tr>
<tr>
<td>Griewank</td>
<td>$f_8(\mathbf{x}) = \frac{1}{4000} \sum_{i=1}^{D} x_i^2 - \frac{1}{D} \sum_{i=1}^{D} \cos \left(\frac{x_i}{\sqrt{D}} \right) + 1$</td>
<td>MM</td>
<td>$[-600, 600]^D$</td>
<td>0</td>
</tr>
</tbody>
</table>

Tab. 1. Standard optimization test functions [42], [43].
<table>
<thead>
<tr>
<th>Benchmark Functions</th>
<th>Algorithm</th>
<th>Best</th>
<th>Average</th>
<th>Median</th>
<th>Standard Deviation</th>
<th>Worst</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>f₁</td>
<td>f₂</td>
<td>f₃</td>
<td>f₄</td>
</tr>
<tr>
<td></td>
<td>ABC</td>
<td>2.4937 × 10⁻¹⁷</td>
<td>7.2440 × 10⁻¹⁷</td>
<td>7.3362 × 10⁻¹⁷</td>
<td>1.8895 × 10⁻¹⁷</td>
<td>1.0211 × 10⁻¹⁶</td>
</tr>
<tr>
<td></td>
<td>MABC</td>
<td>1.9596 × 10⁻¹⁷</td>
<td>5.3625 × 10⁻¹⁷</td>
<td>5.2804 × 10⁻¹⁷</td>
<td>1.6420 × 10⁻¹⁷</td>
<td>8.0280 × 10⁻¹⁷</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>f₃</td>
<td>f₄</td>
<td>f₅</td>
<td>f₆</td>
</tr>
<tr>
<td></td>
<td>ABC</td>
<td>2.7867 × 10⁻⁴</td>
<td>1.5300 × 10⁻²</td>
<td>9.6000 × 10⁻³</td>
<td>1.7100 × 10⁻²</td>
<td>6.4100 × 10⁻²</td>
</tr>
<tr>
<td></td>
<td>MABC</td>
<td>1.2757 × 10⁻⁴</td>
<td>3.4100 × 10⁻²</td>
<td>1.3700 × 10⁻²</td>
<td>5.1000 × 10⁻²</td>
<td>2.3430 × 10⁻¹</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>f₅</td>
<td>f₆</td>
<td>f₇</td>
<td>f₈</td>
</tr>
<tr>
<td></td>
<td>ABC</td>
<td>1.2757 × 10⁻⁴</td>
<td>3.4100 × 10⁻²</td>
<td>1.3700 × 10⁻²</td>
<td>5.1000 × 10⁻²</td>
<td>2.3430 × 10⁻¹</td>
</tr>
<tr>
<td></td>
<td>MABC</td>
<td>2.5746 × 10⁻¹⁷</td>
<td>5.3061 × 10⁻¹⁷</td>
<td>5.3406 × 10⁻¹⁷</td>
<td>1.1987 × 10⁻¹⁷</td>
<td>7.7252 × 10⁻¹⁷</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>f₆</td>
<td>f₇</td>
<td>f₈</td>
<td>f₉</td>
</tr>
<tr>
<td></td>
<td>ABC</td>
<td>5.1404 × 10⁻¹⁷</td>
<td>9.6161 × 10⁻¹⁷</td>
<td>9.1912 × 10⁻¹⁷</td>
<td>3.2381 × 10⁻¹⁷</td>
<td>1.8299 × 10⁻¹⁶</td>
</tr>
<tr>
<td></td>
<td>MABC</td>
<td>2.3582 × 10⁻¹⁷</td>
<td>6.5347 × 10⁻¹⁷</td>
<td>6.7802 × 10⁻¹⁷</td>
<td>1.9038 × 10⁻¹⁷</td>
<td>1.0276 × 10⁻¹⁶</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>f₇</td>
<td>f₈</td>
<td>f₉</td>
<td></td>
</tr>
<tr>
<td></td>
<td>ABC</td>
<td>4.3899 × 10⁻¹⁵</td>
<td>6.8094 × 10⁻¹⁵</td>
<td>7.8894 × 10⁻¹⁵</td>
<td>1.7034 × 10⁻¹⁵</td>
<td>7.8894 × 10⁻¹⁵</td>
</tr>
<tr>
<td></td>
<td>MABC</td>
<td>4.4389 × 10⁻¹⁵</td>
<td>5.6251 × 10⁻¹⁵</td>
<td>4.4409 × 10⁻¹⁵</td>
<td>1.7034 × 10⁻¹⁵</td>
<td>7.9936 × 10⁻¹⁵</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>f₈</td>
<td>f₉</td>
<td>f₁₀</td>
<td></td>
</tr>
<tr>
<td></td>
<td>ABC</td>
<td>0.0000 × 10⁰</td>
<td>2.4654 × 10⁻⁴</td>
<td>0.0000 × 10⁰</td>
<td>1.4000 × 10⁻³</td>
<td>0.0074 × 10⁰</td>
</tr>
<tr>
<td></td>
<td>MABC</td>
<td>0.0000 × 10⁰</td>
<td>3.7007 × 10⁻¹⁷</td>
<td>0.0000 × 10⁰</td>
<td>8.9073 × 10⁻¹⁷</td>
<td>4.4409 × 10⁻¹⁶</td>
</tr>
</tbody>
</table>

Tab. 2. The best, average, median, SD, and worst values obtained by ABC and MABC on benchmark test functions at D = 10.

<table>
<thead>
<tr>
<th>Benchmark Functions</th>
<th>Algorithm</th>
<th>Best</th>
<th>Average</th>
<th>Median</th>
<th>Standard Deviation</th>
<th>Worst</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>f₁</td>
<td>f₂</td>
<td>f₃</td>
<td>f₄</td>
</tr>
<tr>
<td></td>
<td>ABC</td>
<td>1.1409 × 10⁻¹³</td>
<td>1.4039 × 10⁻¹²</td>
<td>1.1574 × 10⁻¹²</td>
<td>1.0900 × 10⁻¹²</td>
<td>5.5826 × 10⁻¹²</td>
</tr>
<tr>
<td></td>
<td>MABC</td>
<td>3.1151 × 10⁻¹⁶</td>
<td>4.4983 × 10⁻¹⁶</td>
<td>4.6959 × 10⁻¹⁶</td>
<td>7.0653 × 10⁻¹⁷</td>
<td>5.3091 × 10⁻¹⁶</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>f₃</td>
<td>f₄</td>
<td>f₅</td>
<td>f₆</td>
</tr>
<tr>
<td></td>
<td>ABC</td>
<td>1.8400 × 10⁻²</td>
<td>4.5720 × 10⁻¹</td>
<td>2.7740 × 10⁻¹</td>
<td>4.8610 × 10⁻¹</td>
<td>2.3057 × 10⁰</td>
</tr>
<tr>
<td></td>
<td>MABC</td>
<td>6.7875 × 10⁻⁴</td>
<td>6.1876 × 10⁰</td>
<td>5.2060 × 10⁻¹</td>
<td>1.4642 × 10⁰</td>
<td>7.3902 × 10⁰</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>f₅</td>
<td>f₆</td>
<td>f₇</td>
<td>f₈</td>
</tr>
<tr>
<td></td>
<td>ABC</td>
<td>3.1631 × 10⁻¹⁴</td>
<td>2.0529 × 10⁻¹³</td>
<td>1.5573 × 10⁻¹³</td>
<td>1.4625 × 10⁻¹³</td>
<td>6.8523 × 10⁻¹³</td>
</tr>
<tr>
<td></td>
<td>MABC</td>
<td>2.9023 × 10⁻¹⁶</td>
<td>4.4656 × 10⁻¹⁶</td>
<td>4.7004 × 10⁻¹⁶</td>
<td>7.0883 × 10⁻¹⁷</td>
<td>5.3520 × 10⁻¹⁶</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>f₆</td>
<td>f₇</td>
<td>f₈</td>
<td>f₉</td>
</tr>
<tr>
<td></td>
<td>ABC</td>
<td>4.9412 × 10⁻¹³</td>
<td>8.8238 × 10⁻¹²</td>
<td>3.8833 × 10⁻¹²</td>
<td>9.4115 × 10⁻¹²</td>
<td>3.5768 × 10⁻¹¹</td>
</tr>
<tr>
<td></td>
<td>MABC</td>
<td>5.0766 × 10⁻¹⁶</td>
<td>6.4164 × 10⁻¹⁶</td>
<td>6.6189 × 10⁻¹⁶</td>
<td>9.4713 × 10⁻¹⁷</td>
<td>8.7348 × 10⁻¹⁶</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>f₇</td>
<td>f₈</td>
<td>f₉</td>
<td>f₁₀</td>
</tr>
<tr>
<td></td>
<td>ABC</td>
<td>2.3010 × 10⁻¹⁰</td>
<td>2.3740 × 10⁻⁶</td>
<td>1.4206 × 10⁻⁹</td>
<td>1.2891 × 10⁻⁵</td>
<td>7.0630 × 10⁻⁵</td>
</tr>
<tr>
<td></td>
<td>MABC</td>
<td>0.0000 × 10⁰</td>
<td>0.0000 × 10⁰</td>
<td>0.0000 × 10⁰</td>
<td>0.0000 × 10⁰</td>
<td>0.0000 × 10⁰</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>f₈</td>
<td>f₉</td>
<td>f₁₀</td>
<td></td>
</tr>
<tr>
<td></td>
<td>ABC</td>
<td>3.8185 × 10⁻⁴</td>
<td>1.5394 × 10⁻²</td>
<td>1.1846 × 10⁻²</td>
<td>8.3483 × 10¹</td>
<td>3.5536 × 10²</td>
</tr>
<tr>
<td></td>
<td>MABC</td>
<td>3.8183 × 10⁻⁴</td>
<td>3.8183 × 10⁻⁴</td>
<td>3.8183 × 10⁻⁴</td>
<td>3.3210 × 10⁻¹³</td>
<td>3.8183 × 10⁻⁴</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>f₉</td>
<td>f₁₀</td>
<td>f₁¹</td>
<td></td>
</tr>
<tr>
<td></td>
<td>MABC</td>
<td>1.2195 × 10⁻¹²</td>
<td>2.4306 × 10⁻¹²</td>
<td>2.3705 × 10⁻¹²</td>
<td>8.0011 × 10⁻¹³</td>
<td>4.2251 × 10⁻¹²</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>f₁₀</td>
<td>f₁¹</td>
<td>f₁²</td>
<td></td>
</tr>
<tr>
<td></td>
<td>ABC</td>
<td>1.3687 × 10⁻¹¹</td>
<td>8.2836 × 10⁻¹⁰</td>
<td>2.9129 × 10⁻¹⁰</td>
<td>1.3363 × 10⁻⁹</td>
<td>5.8847 × 10⁻⁹</td>
</tr>
<tr>
<td></td>
<td>MABC</td>
<td>0.0000 × 10⁰</td>
<td>4.1117 × 10⁻⁴</td>
<td>1.1102 × 10⁻⁶</td>
<td>2.9000 × 10⁻³</td>
<td>1.2300 × 10⁻²</td>
</tr>
</tbody>
</table>

Tab. 3. The best, average, median, SD, and worst values obtained by ABC and MABC on benchmark test functions at D = 30.
Fig. 3. Convergence performance (best solution in each iteration) comparison between MABC and ABC algorithms for eight benchmark test functions ($f_1$ to $f_8$) at different dimensions.
with the total iterations for $D = 10$ and $D = 30$ dimensions. Here, the “Average of the Best Fitness Value” is obtained by computing the mean of the best fitness values of all the runs (i.e., each algorithm repeated for 30 independent runs) in each iteration. The lines in Fig. 3(a)–(h), which do not reach the end of the iterations indicate that the next computation has acquired a zero value. As exhibited in the Fig. 3, except for function $f_2$ with $D = 10$, $30$ and function $f_8$ with $D = 30$, MABC has a swifter convergence speed than the ABC algorithm.

4.2 Performance Evaluation of MABC-based Power Allocation

The MATLAB simulations are shown in this part to assess the effectiveness of the presented MABC-PA algorithm. A single cell with $N = 20$ randomly distributed users is considered in the simulations. Moreover, each sub-channel is paired with the two users based on the adopted pairing scheme. The Rayleigh fading wireless channel model is presumed between BS and users. Further, the BS total transmission power ($P_t = 30$ dBm) is partitioned to all the sub-channels equally. Finally, the proposed MABC-PA is employed to allocate power to paired users of each sub-channel optimally by satisfying all the constraints. We consider the average results based on $10^4$ channel accomplishments. Besides, the parameter values taken for simulation are detailed in Tab. 4. The parameters of the algorithm have been determined after thorough simulations and performance evaluations. We compared the proposed power allocation strategy with ABC-PA, FTPA [25] methods, and OMA scheme.

Figure 4 displays the comparison between the sum throughput and transmit power at the BS for $N = 20$ users. Accordingly, it is clear from Fig. 4 that the sum throughput rises as the transmission power varies. Moreover, at the low transmit powers, the sum rate of MABC-PA is much greater than the FTPA. On the other hand, MABC-PA, ABC-PA, and FTPA methods’ performance are nearer at the higher transmit powers because the power allocation factor for a strong user is saturated for higher values of $P_t$. In other words, the channel gain will become a less significant aspect when the BS transmitted power is more. Further, it can be heeded that due to the modified solution search equation, the proposed MABC-PA exceeds the ABC-PA in terms of achievable sum rate. Besides, we set the minimum capacity for strong and weak users to be 200 kbps and 20 kbps, respectively.

Figure 5 shows the influence of the various users on the total throughput of the system. Here, the transmission power is fixed to 30 dBm. Further, it demonstrates that the system’s total throughput augments as the number of user equipments per BS increases, which signifies that multi-user diversity gain. We can also see that the sum throughput of all NOMA schemes is superior to the OMA due to multiplexing gain when NOMA is used. As a result, Figures 4 and 5 revealed that the proposed MABC-PA is outperformed the ABC-PA, FTPA algorithms, and traditional OMA.

Figure 6 depicts the convergence plot of the MABC-PA and ABC-PA algorithms for NOMA system. From Fig. 6, it is clearly explicit that MABC-PA delivers better performance than the ABC-PA algorithm. Here, the sum rate is averaged over 30 independent runs at $P_t = 30$ dBm for $N = 20$ users. Furthermore, MABC-PA is rapidly converged to an optimal solution compared to the ABC-PA scheme.

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cell radius ($R$)</td>
<td>500 m</td>
</tr>
<tr>
<td>Transmission power ($P_t$)</td>
<td>30 dBm</td>
</tr>
<tr>
<td>Overall bandwidth ($B$)</td>
<td>5 MHz</td>
</tr>
<tr>
<td>Noise density ($N_o$)</td>
<td>$-174$ dBm/Hz</td>
</tr>
<tr>
<td>Number of users ($N$)</td>
<td>20</td>
</tr>
<tr>
<td>Number of sub-channels ($M$)</td>
<td>10</td>
</tr>
<tr>
<td>Path loss exponent ($\theta$)</td>
<td>4</td>
</tr>
<tr>
<td>Min. data rate for $U_E$ ($R_{E\text{max}}$)</td>
<td>200 kbps</td>
</tr>
<tr>
<td>Min. data rate for $U_w$ ($R_{W\text{max}}$)</td>
<td>20 kbps</td>
</tr>
<tr>
<td>Size of the population ($NP$)</td>
<td>50</td>
</tr>
<tr>
<td>Limit</td>
<td>$NP \times D$</td>
</tr>
<tr>
<td>Termination criteria ($T$)</td>
<td>100</td>
</tr>
</tbody>
</table>

Tab. 4. Parameters chosen for simulations.
5. Conclusion

This article studied the unprecedented power allocation approach to enhance the sum throughput for downlink NOMA system, considering BS transmit power and the user’s minimum rate constraints. First, we proposed the Modified ABC approach by improving the solution search equation of “employed and onlooker bees”, thereby enhancing the exploitation in the ABC algorithm. Then, we evaluated every sub-channel user’s optimal powers by employing the proposed MABC optimization method to improve the system’s total throughput. Further, presented MABC-PA algorithm validated with standard benchmark test functions. Moreover, the simulation findings revealed that the MABC-based PA is superior concerning the system’s sum rate to ABC-PA, FTPA, and traditional OMA. In addition, the proposed MABC algorithm has converged faster than the standard ABC algorithm.

In the future, we plan to implement the multi-user downlink NOMA system with the proposed MABC-based power allocation strategy utilizing the software-defined radio platform. Moreover, this work could be expanded to include additional performance metrics for the problem mentioned above, as well as the implementation of the MIMO-NOMA scenario in order to improve the system’s performance further.

References


SUNKARABOINA SREENU
received the B.Tech degree from JNTUH and the M.Tech degree from Acharya Nagarjuna University, India, in 2012 and 2014, respectively. He is presently pursuing Ph.D. in the National Institute of Technology, Warangal, India. His current research interests include 5G wireless networks, non-orthogonal multiple access, and optimization theory.

NAIDU KALPANA (corresponding author) received the Ph.D. degree from IIT-Hyderabad, India, in Jan, 2016. Since 2018 she has been working as an Assistant Professor with the National Institute of Technology (NIT), Warangal. Her research interests include the resource allocation in 6G wireless networks, heterogeneous networks, etc. She presided over technical sessions at international conferences as a chair.