Mathematical Biosciences and Engineering

MBE, 20(8): 14675–14698.
DOI: 10.3934/mbe.2023656
Received: 15 March 2023
Revised: 06 June 2023
Accepted: 07 June 2023
Published: 06 July 2023

http://www.aimspress.com/journal/MBE

Research article

Reliable task allocation for soil moisture wireless sensor networks using differential evolution adaptive elite butterfly optimization algorithm

Haitao Huang[1], Min Tian[1,*], Jie Zhou[2] and Xiang Liu[1]

[1] College of mechanical and electrical engineering, Shihezi University, Shihezi 832000, China
[2] College of information science and technology, Shihezi University, Shihezi 832000, China

* **Correspondence:** Email: tm_inf@shzu.edu.cn; Tel: +8613319937952.

Abstract: Wireless sensor technology advancements have made soil moisture wireless sensor networks (SMWSNs) a vital component of precision agriculture. However, the humidity nodes in SMWSNs have a weak ability in information collection, storage, calculation, etc. Hence, it is essential to reasonably pursue task allocation for SMWSNs to improve the network benefits of SMWSNs. However, the task allocation of SMWSNs is an NP (Non-deterministic Polynomial)-hard issue, and its complexity becomes even higher when constraints such as limited computing capabilities and power are taken into consideration. In this paper, a novel differential evolution adaptive elite butterfly optimization algorithm (DEAEBOA) is proposed. DEAEBOA has significantly improved the task allocation efficiency of SMWSNs, effectively avoided plan stagnation, and greatly accelerated the convergence speed. In the meantime, a new adaptive operator was designed, which signally ameliorates the accuracy and performance of the algorithm. In addition, a new elite operator and differential evolution strategy are put forward to markedly enhance the global search ability, which can availably avoid local optimization. Simulation experiments were carried out by comparing DEAEBOA with the butterfly optimization algorithm (BOA), particle swarm optimization (PSO), genetic algorithm (GA), and beluga whale optimization (BWO). The simulation results show that DEAEBOA significantly improved the task allocation efficiency, and compared with BOA, PSO, GA, and BWO the network benefit rate increased by 11.86%, 5.46%, 8.98%, and 12.18% respectively.

Keywords: task allocation; precision agriculture; soil moisture wireless sensor networks; algorithm; network benefit

1. Introduction

The successful application of wireless sensor networks in monitoring, spectrum management, distributed detection, agriculture, and other fields is very important for a country's development [1–3]. As the underlying supporting technology of the Agricultural Internet of Things (AIoT), wireless sensor networks are widely deployed in various agricultural applications [4,5]. Under the new development direction of world agriculture, precision agriculture is gradually replacing traditional agriculture [6]. Soil moisture wireless sensor networks (SMWSNs) play a critical role in enabling precise irrigation of farmland, and SMWSNs are an essential tool for agricultural production [7,8]. SMWSNs have the characteristics of low cost, flexible structure, self-organizing network, and dynamic topology, which can accurately collect and effectively transmit soil moisture data and subsequently control the terminal to accurately adjust the amount of farmland irrigation [9,10]. By deploying SMWSNs in field conditions, farmers can promptly make more accurate agricultural production decisions based on monitoring the soil moisture information collected by sensor nodes in real time [11–13]. Figure 1 shows the sensor node information transmission. Task allocation is a difficult issue in wireless sensor networks. In wireless sensors, it is often necessary to perform various tasks, such as computing tasks, perception tasks, and communication tasks [14]. Task allocation reflects that in the dynamic environment of the network with limited energy, the tasks in the system are effectively allocated first, and then the specified tasks are scheduled to be executed on suitable nodes to achieve optimal performance [15]. Optimizing the task allocation of SMWSNs is of great significance for rationally allocating network resources to better complete information acquisition and precision irrigation [16]. SMWSNs are broadly applied in crop irrigation, though they also face the problems of a complex external environment and limited network resources. These problems will lead to untimely information collection and transmission, reducing the overall performance of the network, and thereby affecting agricultural production.

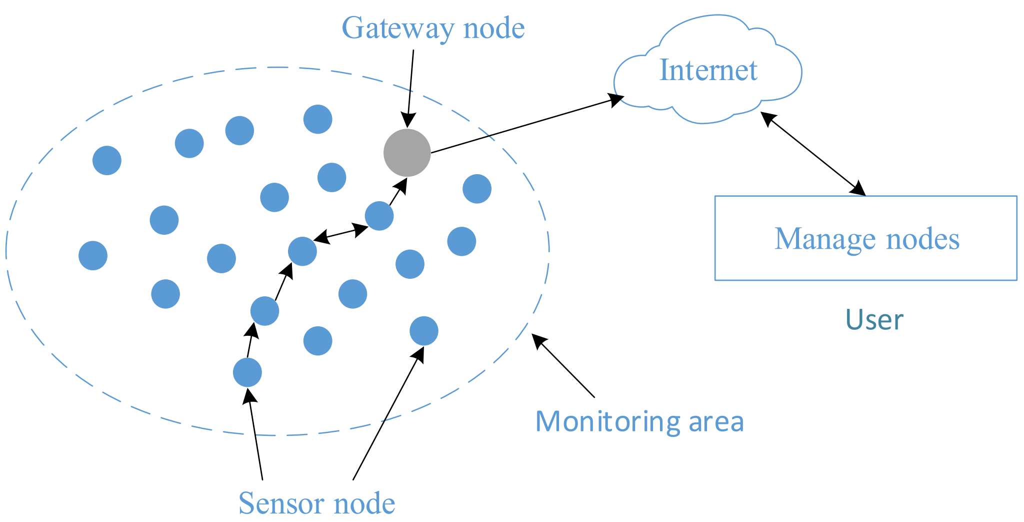

Figure 1. The sensor node information transmission.

The problem of wireless sensor task allocation optimization has always been a hot topic discussed by researchers. Scholars have attempted numerous analyses and research on heuristic algorithms to address this NP-hard problem [17]. For example, Issac et al. [18] proposed a method using particle swarm optimization (PSO) to solve the multi-objective task allocation problem, which is more efficient than the task allocation of traditional PSO. The generation of the artificial bee colony algorithm (ABC) is inspired by the foraging activities of bees to seek out the optimal solution to the issue [19], where different activities are carried out according to their respective division of labor to achieve information sharing and communication. Hou et al. [20] designed an improved genetic algorithm (IGA) for the shortcomings of GA, which optimizes the task allocation problem by producing offspring to inherit the best characteristics of the previous generation. Shin et al. [21] designed an improved ant colony optimization algorithm (IACO) inspired by ant foraging behavior in nature. Gao et al. [22] proposed an improved firefly algorithm (IFA) based on simulated annealing and adaptive mutations to resolve task allocation problems, which have a better optimization ability and higher solution accuracy compared with FA. Although these heuristics can improve the efficiency of task allocation to some extent, they have the problem of low availability in evolutionary stagnation and local optimization. Applying it to the task problem of SMWSNs will lead to premature convergence and reduce network revenue. Chen et al. proposed an algorithm based on deep learning and developed a new activation function, the parameter-free adaptive Swish (PASwish), which can perform a more flexible nonlinear transformation of different input data; however, this algorithm is of a high complexity [23]. This paper mainly proposes DEAEBOA to obtain an improved higher network benefit and finally obtain an efficient task allocation scheme. The superiority of DEAEBOA is reflected by assigning different tasks to either different or the same number of nodes, saving sensor node resources to the greatest extent, and finally maximizing network benefits from task allocation.

In solving the problem of revenue maximization, DEAEBOA has an outstanding performance in SMWSNs, vastly heightening the overall benefit of the network. This paper has the following main contributions:

1) First, a novel DEAEBOA is proposed. Compared with the aforementioned studies, the algorithm can greatly enhance the efficiency of task allocation for SMWSNs. Meanwhile, it avoids evolutionary stagnation and significantly accelerates the convergence speed, which improves the performance of SMWSNs and effectively saves network resources; on the other hand, the method realizes the precise adjustment of farmland irrigation amount within the actual environment.

2) Second, a new adaptive operator is designed. Compared with the aforementioned studies, the algorithm can immensely improve the accuracy and performance. Moreover, a novel elite operator and differential evolution strategy are designed, which can efficaciously enhance the global search capacity and avert falling into local optimization.

3) Thirdly, a new mathematical model for task allocation optimization of SMWSNs is established. Compared with recent model studies, the algorithm can consider and optimize several fundamental factors affecting the performance of SMWSNs, including node sensing distance, number of nodes, and information acquisition advantages. In addition, the model employs a new objective function to balance the relationship between these factors and serve as an evaluation criterion for network benefits.

The general framework of this article is as follows. Section 2 covers the assignment of tasks related to SMWSNs. Section 3 details the shared task assignment model. Section 4 recommends DEAEBOA address task allocation in SMWSNs. Subsequently, the validity of DEAEBOA's results through simulation experiments is discussed in Section 5. Section 6 returns to the conclusion.

2. Relate work

Wireless sensor networks have good stability, which is important for the time delay and security of signal transmission [24,25]. Heuristics are widely used in agricultural wireless sensor networks. In order to provide a fuller introduction to the work related to the assignment of tasks related to SMWSNs, this chapter will be discussed in two parts. First, the necessity of heuristic algorithms in the task allocation of agricultural wireless sensors is introduced. Finally, the application of heuristic algorithms in wireless sensors is probed. In the meantime, the innovative heuristic algorithm in this paper is elaborated.

2.1. The necessity of the heuristic algorithm for WSNs in agricultural

In the period of agricultural modernization, the task distribution of wireless sensors in agriculture has attracted the attention of many scholars [26]. The task assignment problem is a typical multi-objective math issue, the purpose of which is to further optimize the performance of the system by designing a reasonable and effective allocation scheme to optimize the execution effect of all tasks [27]. According to the environment and resource constraints, a mathematical model of task allocation is established, and an objective function is also required to use efficient algorithms to solve the optimal task allocation scheme [28]. The methods for solving task allocation problems are mainly mathematical programming methods and heuristic algorithms. The mathematical programming method has evident advantages in solving constraint problems with a small number of constraints and variables. However, the agricultural production environment is complex and changeable, resulting in an increase in the amount of system network calculation, a low solution efficiency, and an inability to adapt to the development environment of smart agriculture [29]. Therefore, traditional mathematical methods to solve task allocation problems in SMWSNs are no longer desirable. Due to the advantages of simple and easy implementation, fast calculation speeds, and strong compatibility, the heuristic algorithm is widely used in agricultural wireless sensors. Compared with mathematical programming, heuristic algorithms can handle task allocation problems in SMWSNs well [30,31].

2.2. Clinical trial registration the use of heuristic algorithms in WSNs

In recent research on task assignment in wireless sensor networks, many scholars have proposed their own solutions. Okhovvat et al. [32] proposed an analytical method based on the queuing theory to calculate the optimal assignment rate from task to actor node. To solve the task assignment problem in virtual wireless sensor networks, Raee et al. [33] designed a scalable heuristic algorithm that minimizes total energy consumption. Kori et al. [34] proposed a classified regression tree machine learning algorithm (CART) for resource allocation schemes to improve the performance and extend the service life of wireless sensor networks by adopting intelligent resource management schemes. Baniabdelghany et al. [35] designed an offline discrete particle swarm optimization (DPSO-TA) algorithm for reliable task assignment in wireless sensor networks of the Internet of Things (IoT), which effectively reduces network energy consumption. However, the algorithms mentioned above have some problems, such as large computation and slow convergence.

To more intuitively reflect the superiority of DEAEBOA, the algorithm proposed by other researchers mentioned below will be compared in three key points, as shown in Table 1.

The earliest heuristic algorithms proposed were GA, PSO, and ACO, in which these bionic algorithms attracted the favor of researchers once proposed. In-depth research based on these traditional heuristics led to the proposal of many new heuristics to solve task allocation problems. Li et al. [36] proposed an improved adaptive cloning genetic algorithm (IACGA) and achieved good results, though it was prone to evolutionary stagnation. Baniabdelghany et al. [35] proposed a discrete particle swarm optimization (DPSO) for distributing tasks between sensor nodes. This method saves time and cost and improves accuracy, but it is effortless to fall into local optimization. Xu et al. [37] designed an elite immune ant colony optimization (EIACO) to dispose task allocation problems. The algorithm has a better task allocation efficiency and higher convergence velocity, though the algorithm complexity is high and the performance is poor.

Sensor nodes can collaboratively perform complex tasks but are often constrained by energy and capacity. In this case, it is critical to find a task allocation scheme that significantly improves the efficiency of the network. Niccolai et al. [38] designed a new evolutionary algorithm for social network optimization (SNO) to deal with the task allocation trouble, which effectively improves the task assignment efficiency under the condition of energy and energy constraints; however, the algorithm has the matter of high computational complexity. Weikert et al. [39] proposed a multi-objective optimization algorithm (MOA) for the dynamic task allocation problem of the IoT. Compared with the existing single-objective algorithm, this method reduces the network latency and improves the network performance. However, with additional tasks and nodes, the computational complexity of the method increases too quickly. Arora et al. [40] used a butterfly optimization algorithm (BOA) to work out problems in wireless sensors, and achieved relatively good performance. However, one problem with this algorithm is that it tends to converge prematurely.

Table 1. Comparison of three key points of the algorithm.

Algorithm	Rate of convergence	Local optimization	Computational complexity
DEAEBOA	fast	\	low
IACGA	fast	√	low
DPSO	fast	√	low
EIACO	fast	√	high
SNO	fast	\	high
MOA	fast	\	high
BOA	fast	√	high

Due to aforementioned methods for solving wireless sensor task allocation, there are problems such as high computational complexity and an ease of falling into local optimization, resulting in a high time delay, waste of network resources, and low task allocation efficiency. Therefore, it is necessary to design a strategy that can avoid premature convergence and has a low computational complexity. This paper proposes a novel DEAEBOA. DEAEBOA has the characteristics of fast convergence speed and low complexity. In SMWSNs, there is a certain relationship between the collection advantages of soil moisture information collection tasks and network benefits, and DEAEBOA can obtain a task allocation scheme that maximizes network benefits. Compared with other heuristic methods, it is easier to jump out of local optimization, which can obviously elevate the task allocation efficiency of SMWSNs and save network resources. In addition, a unique task allocation model for SMWSNs is proposed, which has the preponderances of low power consumption and simple processing. DEAEBOA optimizes the SMWSNs model to improve task allocation efficiency and network performance. A simplified diagram of DEAEBOA acting on SMWSNs is shown in Figure 2.

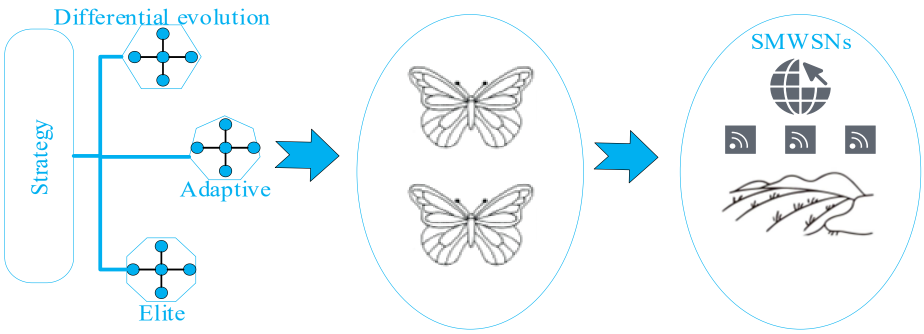

Figure 2. DEAEBOA acting on SMWSNs.

3. System model

3.1. Problem statement

Table 2. Explanation of symbols.

Notation	Statement
m	Number of soil moisture information collection tasks
n	Number of sensor nodes
B	Soil moisture information collection advantage matrix
U	Economic benefit matrix
D	Solution in integer form
Z	Solution in binary form
Z_b	Soil moisture information collection advantage matrix specific solutions
F	Soil moisture information collection benefit matrix
benefit	Total soil moisture information collection benefit

In SMWSNs, sensor nodes are deployed in farmland areas to monitor soil moisture conditions. In agricultural production, it is more necessary to pay attention to the synergy and economic benefit between sensor nodes. Therefore, to enhance the efficiency of SMWSNs and to guarantee the stable performance of sensor networks, an excellent distribution scheme should be adopted to allocate a wide range of tasks to a limited number of sensor nodes. In addition, the symbol descriptions that appear in this section are listed in Table 2.

3.2. Soil moisture information collection task allocation model

In this paper, an efficient task allocation model based on SMWSNs is designed. In SMWSNs, the model can be simply described as assigning *m* soil moisture information collection tasks to *n* soil moisture sensor nodes according to the economic benefit and information collection advantages. Visually, Figure 3 depicts the task assignment model.

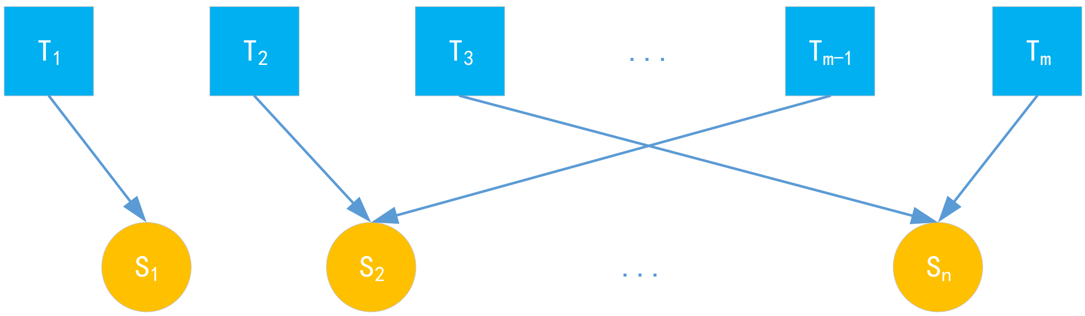

Figure 3. Soil moisture information collection task allocation model.

Due to the difference in the distance between each sensor node and each farmland area, the advantages of different sensor nodes collecting soil moisture information in different areas may also be different. The predominance matrix is shown in (1):

$$B=\begin{bmatrix} b_{1,1} & b_{1,2} & \cdots & b_{1,m-1} & b_{1,m} \\ b_{2,1} & b_{2,2} & \cdots & b_{2,m-1} & b_{2,m} \\ \cdots & \cdots & b_{x,y} & \cdots & \cdots \\ b_{n-1,1} & b_{n-1,2} & \cdots & b_{n-1,m-1} & b_{n-1,m} \\ b_{n,1} & b_{n,2} & \cdots & b_{n,m-1} & b_{n,m} \end{bmatrix}(x\in[1,n],y\in[1,m]), \tag{1}$$

where $\mathrm{b_{xy}}$ represents the soil moisture information collection advantage of the $\mathrm{x_{th}}$ humidity sensor node over the y_{th} farmland area, which is equal to the reciprocal of the distance between the y_{th} farmland area and the x_{th} humidity sensor node. The closer the distance, the better the soil moisture information collected. At the same time, the larger the b_{xy}, the closer the x_{th} humidity sensor node is to the y_{th} area, and the greater the soil moisture information collection advantage of the x_{th} humidity sensor node for the y_{th} farmland area. Suppose that the advantages of farmland soil moisture information collection are evaluated and ranked, and the economic benefit is evaluated and described as (2):

$$U=[u_1 \quad u_2 \quad \dots \quad u_y \quad \dots \quad u_{m-1} \quad u_m](y\in[1,m]), \tag{2}$$

where u_y represents the economic benefit of the j_{th} farmland area. Then, the allocation scheme can be represented as

$$D=[d_1 \quad d_2 \quad \dots \quad d_k \quad \dots \quad d_{m-1} \quad d_m](k\in[1,m],d_k\in[1,n]). \tag{3}$$

In (3), the number in D indicates which sensor node is assigned to, and D is used to store the allocation result. Section 4 describes coding schemes and particle initialization. In addition, in order to facilitate subsequent operations, the allocation scheme in binary form is given as follows:

$$Z = \begin{bmatrix} z_{1,1} & z_{1,2} & z_{1,3} & \cdots & z_{1,m} \\ z_{2,1} & z_{2,2} & z_{2,3} & \cdots & z_{2,m} \\ \cdots & \cdots & \cdots & z_{i,j} & \cdots \\ z_{n,1} & z_{n,2} & z_{n,3} & \cdots & z_{n,m} \end{bmatrix} \left(z_{i,j} \in \{0,1\}, i \in [1,n], j \in [1,m]\right). \tag{4}$$

In (4), Z represents a randomly generated binary task allocation scheme. The quantity of rows in Z corresponds to the quantity of soil moisture sensor nodes, and the quantity of columns corresponds to the quantity of soil moisture information collection tasks. $z_{i,j} = 1$ indicates that the i_{th} sensor node is responsible for the j_{th} soil moisture information acquisition task; otherwise, the opposite is true.

For example, let's take a model with 5 soil moisture information collection tasks and 3 soil moisture sensor nodes. As shown in Figure 4, the soil moisture information collection task allocation model.

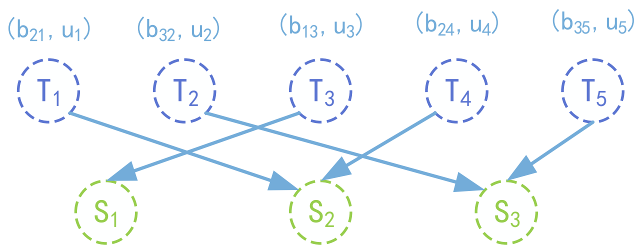

Figure 4. A soil moisture information collection task allocation model.

$$B = \begin{bmatrix} b_{11} & b_{12} & b_{13} & b_{14} & b_{15} \\ b_{21} & b_{22} & b_{23} & b_{24} & b_{25} \\ b_{31} & b_{32} & b_{33} & b_{34} & b_{35} \end{bmatrix} \tag{5}$$

$$U = [u_1 \quad u_2 \quad u_3 \quad u_4 \quad u_5] \tag{6}$$

In (5) and (6), the number of columns in B and U corresponds to the number of soil moisture information collection tasks, while the quantity of rows in B corresponds to the quantity of soil moisture sensors. For clarity, the allocation scheme for the example model is shown below:

$$D = [2 \quad 3 \quad 1 \quad 2 \quad 3], \tag{7}$$

$$Z = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 \end{bmatrix}, \tag{8}$$

where D and Z represent the allocation scheme in integer form and in binary form, respectively.

$$Z_b = B \cdot Z \tag{9}$$

$$Z_b = \begin{bmatrix} 0 & 0 & b_{13} & 0 & 0 \\ b_{21} & 0 & 0 & b_{24} & 0 \\ 0 & b_{32} & 0 & 0 & b_{35} \end{bmatrix} \tag{10}$$

In (9), Z_b represents the advantage matrix corresponding to the randomly generated soil moisture information collection task allocation scheme *Z*. In addition, the value in Z_b represents the advantage of j_{th} soil moisture information collection task being allocated to the i_{th} soil moisture sensor node.

$$F = Z_b \times W^T \tag{11}$$

Finally, the soil moisture information acquisition benefit matrix *F* is calculated in (11).

$$benefit = \sum_{i=1}^{n} f_i \tag{12}$$

In (12), f_i represents the value of i_{th} in the benefit matrix *F*, and the total soil moisture information collection benefit of the entire system can be obtained by calculating the sum of all f_i.

In SMWSNs, the task allocation scheme not only realizes the precise irrigation of farmland, but also increases the actual economic value of farmland and enormously elevates the efficiency of the system. In this way, each sensor node has a certain benefit in completing the corresponding task, and the sum of the benefit of all nodes is the total benefit of the whole system. Due to the influence of environmental factors and the complexity of practical problems in agricultural production, it is also necessary to consider the performance of sensor nodes and the load difference of the task assigned.

4. DEAEBOA for task allocation optimization in SMWSNs

In order to make SMWSNs have a better network benefit, a task allocation method based on DEAEBOA is proposed. This method possesses a sturdy global search ability and can stably converge. At the same time, due to the innovative introduction of adaptive operators, differential evolution, and elite strategies, the method can effectively avoid falling into local optimum and premature convergence. Moreover, these strategies enable DEAEBOA to obtain the optimal information collection task assignment scheme and significantly improve the network benefit of SMWSNs. DEAEBOA is an evolutionary algorithm that mimics how butterflies search for food. In DEAEBOA, the optimization process of the problem is seen as a group of scented butterflies attracting each other. Each butterfly, also known as a particle, corresponds to a potential solution to a problem. Then, finding the particle with the strongest fragrance is the optimal solution to the problem. Initially, this group of particles emits different amounts of fragrance, each either moving randomly or towards the particles that emit more fragrance. Over the course of each iteration, each particle is continuously adjusted according to its optimal position and the optimal position of the population, recalculating the particle position and fragrance intensity to achieve the best results.

The process of DEAEBOA can be described as coding schemes and particle initialization, fitness calculations, particle motion, differential evolution strategies, adaptive operators, elite operators, and termination conditions.

4.1. Encoding scheme and particle initialization

Encoding is the first step in DEAEBOA. There are two commonly used encoding methods: one is binary encoding and the other is integer encoding. In order to optimize the network benefits of SMWSNs as much as possible, as well as to perform a wider range of searches, this article uses integer encoding. Assuming that there are *x* particles, n soil moisture sensors, and m tasks in the initial group, it can be described as (13). $D_1, D_2 \cdots D_x$ are the *x* particles in the initial group, each of which is a vector of length *m*, representing a potential solution. The values in the vector represent integers between 1 and n, which is consistent with Eq (3). First, the initial array is randomly generated, rounding the initial group to limit the values to a range of 1 to n. Each time the group is updated, a rounding operation is made to limit the value to a range of 1 to n.

$$group = \begin{bmatrix} D_1 \\ D_2 \\ D_3 \\ \vdots \\ D_x \end{bmatrix} = \begin{bmatrix} d_{1,1} & d_{1,2} & d_{1,3} & \cdots & d_{1,m} \\ d_{2,1} & d_{2,2} & d_{2,3} & \cdots & d_{2,m} \\ d_{3,1} & d_{3,2} & d_{3,3} & \cdots & d_{3,m} \\ \cdots & \cdots & \cdots & d_{u,v} & \cdots \\ d_{x,1} & d_{x,2} & d_{x,3} & \cdots & d_{x,m} \end{bmatrix} \tag{13}$$

4.2. Fitness calculation

In general, a large factor influencing the convergence of DEAEBOA is the objective function. Each particle has its corresponding value and a corresponding degree of fitness. In DEAEBOA, fitness is assessed using the Eq (12). According to the above Eqs (11) and (12), it can be seen that in order to achieve the maximum benefit of the network, the *benefit* needs to be as large as possible, that is, Z_b needs to be as large as possible.

With each particle iteration, its fitness is compared to its historical optimal solution. If the current particle fitness is high, the particle historical optimal solution is updated, which is the current particle fitness. Otherwise, leave it as is. Second, the current particle fitness is compared with the global optimal solution of this group. If the current particle fitness is high, the global optimal solution of the particle is updated and replaces the current optimal solution position; otherwise, it will not change.

4.3. Particle movement

In particle initialization, the initial position of each particle is different. Each particle emits odors of different intensities depending on where it is located. The odor formula is described in (14). Depending on the effect of odor intensity, each particle either moves randomly or towards particles that emit more odors. When a particle moves towards a particle with a strong odor concentration, it is called a global search. The equation is described in (15). Instead, the particles move randomly, which is a process called local search. The equation is described in (16). Conversion probability p to control the global and local search process, $p \in [0, 1]$.

$$f = c^{\mathrm{t}} I^{\mathrm{a}} \tag{14}$$

$$x_i{}^{t+1} = x_i{}^t + (r^2 \times g^* - x_i{}^t) \times f_i \tag{15}$$

$$x_i{}^{t+1} = x_i{}^t + (r^2 \times x_j{}^t - x_k{}^t) \times f_i \tag{16}$$

In Eq (14), f is the amount of fragrance perception and c is the sensory form. I represents the stimulus intensity, that is, the fitness function value, and a stands for fragrance, $a, c \in [0,1]$. c^t represents the form of sensation in the t generation. In Eq (15), $x_i{}^{t+1}$ represents the position of the i_{th} individual in the $t+1$ generation, $r \in [0,1]$, g^* represents the global optimal solution, f_i represents the scent perception of the i_{th} individual. In Eq (16), $x_i{}^t, x_j{}^t, x_k{}^t$ indicates the position of the I, j, and k individuals in the t generation, respectively.

To be more in line with the laws of nature, the value of c will change accordingly during each iteration of the particle. The specific operation is as follows:

$$c^{t+1} = c^t + (b/(c^t \times Ngen)) \tag{17}$$

where c^{t+1} represents the form of sensation in the t generation, b is a constant, and $Ngen$ is the maximum number of iterations.

4.4. Differential evolution strategy

The differential evolution strategy consists of three parts: mutation, crossover, and selection operations. Finally, the particles with the highest adaptability are retained. The specific operation is as follows.

4.4.1. Mutation operation

In each iteration, the mutation process is to select three different particles in the group and generate new particles through Eq (18):

$$V_{i,G+1} = X_{r3,G} + F * (X_{r1,G} - X_{r2,G}) \tag{18}$$

where i, r_1, r_2, and r_3 belong to positive integers from 1 to n, $i \neq r_1 \neq r_2 \neq r_3$, and F is the variation probability, $F \in [0,1]$.

4.4.2. Cross operation

We can select according to a certain probability, and the new particle produced in Eq (18) crosses the current particle; otherwise, leave it as it is:

$$U_{j,i,G+1} = \begin{cases} v_{j,i,G+1} & if\ j_{rand} \leq CR \\ x_{j,i,G} & otherwise \end{cases} \tag{19}$$

where CR is the crossover probability factor and j_{rand} is a random component, ensuring that at least a one-dimensional component of the new particle after the crossing is provided by the mutant particles.

4.4.3. Selection operation

Based on the fitness function value, the better of the new particles and target particles that have undergone mutation and cross-generation is selected as the next generation.

$$X_{i,G+1} = \begin{cases} U_{i,G+1} & if\ f(U_{i,G+1}) \leq f(X_{i,G}) \\ X_{i,G} & otherwise \end{cases} \tag{20}$$

4.5. Adaptive operators

Adaptive operators have obvious advantages in dealing with complex optimization problems. In previous classical evolutionary algorithms, scholars typically set constants that remain constant throughout the lifetime of the algorithm. One advantage of this is that it can decrease the challenge of algorithm design, making the algorithm less complex; however, the algorithm tends to converge prematurely and fails to achieve the expected results. The adaptive operator designed in this paper is used to control a single parameter. We set the conversion probability in the algorithm to *P*. The adaptive conversion probability *p* controls whether particles are searched either globally or locally. If the setting is too large, the particles will not conform to the natural evolution law and the algorithm will be unstable. If the setting is too small, the evolution speed of the algorithm will be affected. Hence, we adopt an adaptive strategy for *P* to enhance the performance of the algorithm. For the case where *P* is close to zero, resulting in premature convergence of the algorithm, we design Eq (21) to quantify the conversion probability under nonlinear changes in individual fitness values:

$$P(i) = P_{min} + Q * (P_{max} - P_{min}) * \frac{f_{max} - f(i)}{f_{max} - f_{min}} \tag{21}$$

where P_{max} and P_{min} are the upper limit of 1 and the lower limit of 0.7, respectively, Q is set to 0.92, f_{max} represents the value with the greatest fitness of all particles, f_{min} is the minimum value of all particles, and $f(i)$ represents the current fitness of the i_{th} particle.

4.6. Elite operators

In order to shorten the search time, this paper designs an elite operator that proposes that the best particles of the previous generation replace the worst particles of the next generation. This makes the algorithm converge faster.

4.7. Conditions of termination

During each iteration, DEAEBOA checks whether the termination conditions are met. The termination condition of the algorithm is to reach the maximum number of iterations.

4.8. The procedure of DEAEBOA

In the algorithm flow shown in Figure 5, we will describe the algorithm flow in detail.

Step 1. Initialize the parameters in DEAEBOA, set the maximum quantity of iterations, cross probability, variation probability, fragrance parameter, sensory form parameter, maximum and minimum transition probability, and randomly generate the initial group. According to Eq (1), the sensor node information acquisition advantage matrix is generated. The task urgency matrix is generated by Eq (2).

Step 2. Through Eqs (11) and (12), the fitness of the old group is obtained, and the historical optimal value of the particle and the global optimal value of the group are recorded.

Step 3. Use Eq (14) to calculate the perceived amount of fragrance, and use Eqs (15) and (16) to search either globally or locally. Renew the sensory form according to Eq (17).

Step 4. To be more in line with the laws of nature, a differential evolution operation is carried out.

Step 5. Equations (11) and (12) are used to assess the fitness of a new swarm and to record the historical optimal of the particle and the global optimal of the swarm.

Step 6. Use an adaptive strategy for the conversion probability according to Eq (21).

Step 7. Execute elite strategies.

Step 8. Repeat 3 through 7 until the termination condition is met, that is, the maximum quantity of iterations is reached.

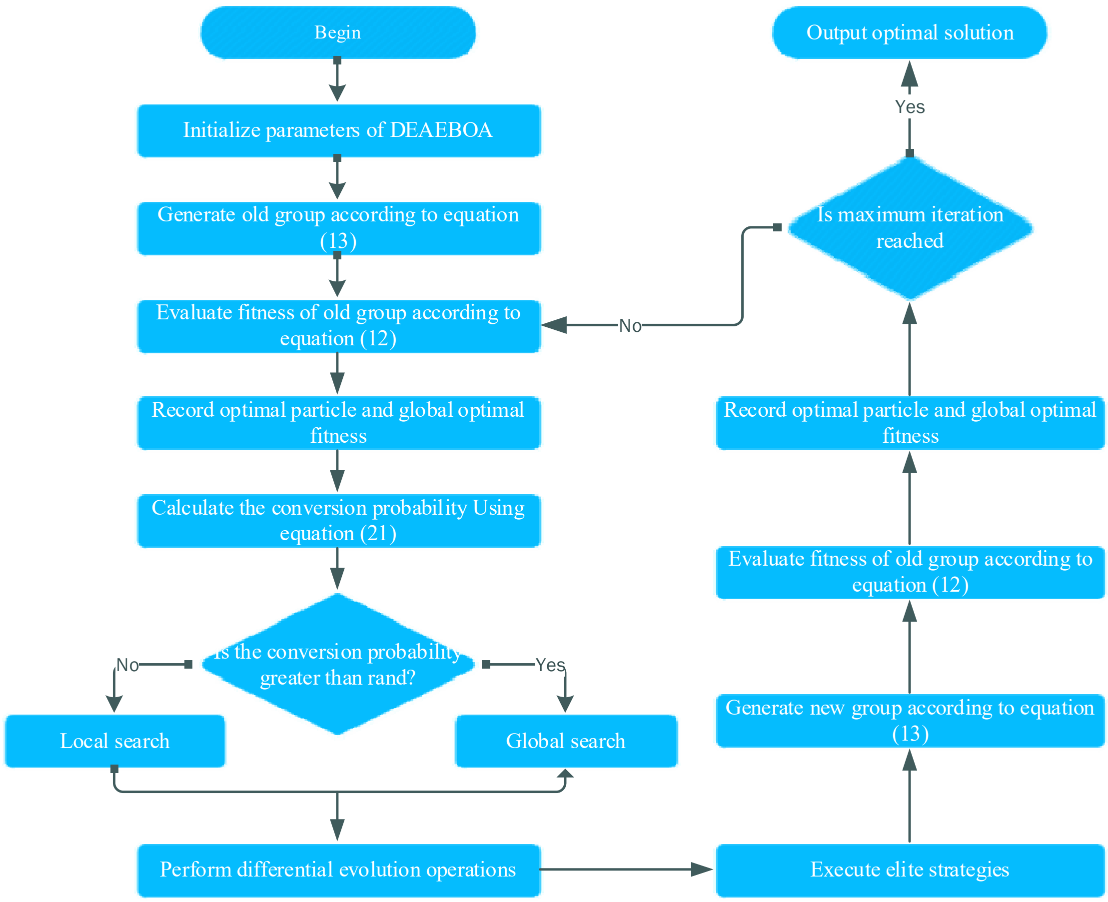

Figure 5. DEAEBOA execution step diagram.

4.9. Difficulties in analytical methods

The difficulty of this analysis method lies in DEAEBOA. When the position of the particle is updated, the adaptive conversion probability is designed, which changes with the change of the number of iterations, which increases the difficulty of the algorithm, and conforms to the random law of natural change. In order to improve the convergence speed of the algorithm, the elite strategy is adopted to ensure the stability of the whole simulation experiment. Finally, the variation and selection operation of the differential evolution can avoid the local optimal situation, but increase the difficulty of the algorithm. The organic combination of the three strategies improves the performance of SMWSNs.

4.10. DEAEBOA temporal complexity analysis

Suppose the population size is N, the number of iterations is MAX_GEN, and the dimension is D. Based on the operation rules of the time complexity symbol O, it can be deduced that the time complexity of DEAEBOA randomly initializing the population is $O\ (N \cdot D)$, and the time complexity of finding the local optimal is O ($N{\cdot}D$). The time complexity of global position update change using differential evolution adaptive elite operator strategy is O ($MaxIter{\cdot}N{\cdot}D$), and the total time complexity of DEAEBOA is O ($MaxIt{\cdot}N{\cdot}D$). Therefore, the time complexity of DEAEBOA is the same as that of BOA, and the computation work will not be increased.

5. Simulation and discussion

5.1. Experimental setup

In this section, DEAEBOA's performance in soil moisture information collection task allocation will be tested through a simulation to achieve the precise irrigation of farmland. In simulation, DEAEBOA is compared with PSO, GA, and BOA methods in different situations, and the average of 100 experiments is used as the result of the four algorithms. The task allocation model with five different algorithms is simulated on a computer with an Intel(R) Core (TM) i7-7700H CPU, 8GB RAM, and Windows 10 operating system by using MATLAB R2022a.

Table 3. Main experimental parameters of the five algorithms.

DE-AEBOA	Population size	Adaptive conversion probability	Mutation probability	Crossover probability	Fragrance	Sensory form
	100	0.7–0.9	0.95	0.95	0.1	0.01
BOA	Population size	Conversion probability	Fragrance	Sensory form		
	100	0.95	0.1	0.01		
PSO	Population size	Inertia weight	Self-learning factor	Group-learning factor		
	100	0.5	0.6	0.6		
GA	Population size	Mutation probability				
	100	0.1				
BWO	Population size	Equilibrium factor				
	100	0.5				

During the simulation experiment, for the accuracy of the simulation, and to compare the performance of DEAEBOA with BOA, GA, PSO, and BWO, we set the population size of the five algorithms to 100, and the maximum quantity of iterations in this paper was set to 800 and 150. In DEAEBOA, we set the adaptive conversion probability range to 0.7–0.9, the mutation probability to 0.95, and the crossover probability to 0.95. In BOA, we set the conversion probability to 0.9. In GA, we set the mutation probability to 0.4. In PSO, we set the inertia weight to 0.5, and both the self-learning factor and the group-learning factor to 0.55. For DEAEBOA, BOA, GA, and PSO, Table 3 shows the main parameters of the four algorithms.

5.2. Discussion of experimental results

To make the simulation results more scientific and researchable, in the discussion section of the experimental results, the data will be analyzed from multiple angles, displayed by simulating evolution curves, line charts, bar charts, pie charts and tables, and at the same time the charts are analyzed and discussed.

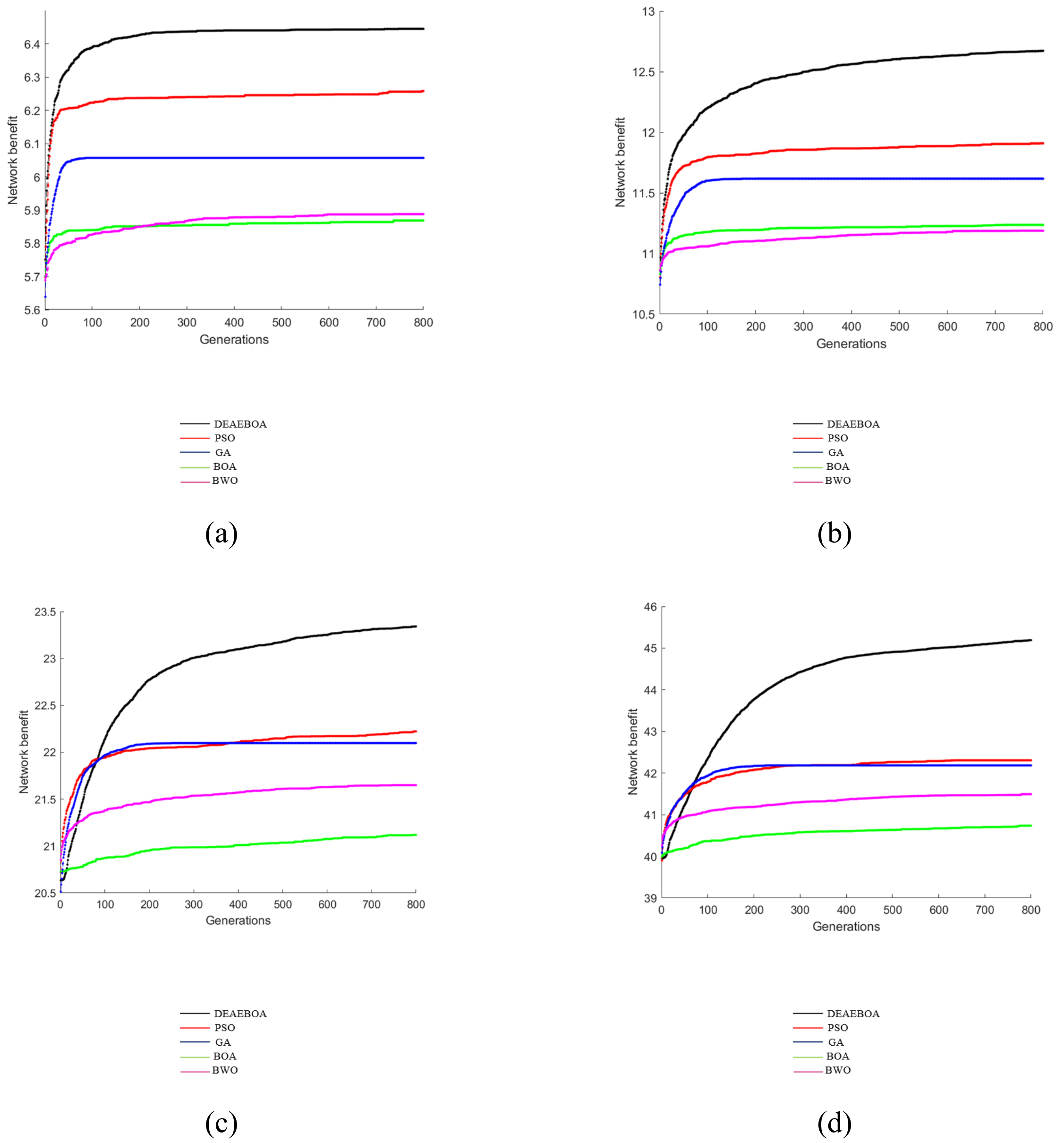

Figure 6. Comparison of network benefit optimization trends of five algorithms in 800 iterations: (a) the benefit of assigning 15 tasks to 8 nodes; (b) the benefit of assigning 30 tasks to 16 nodes; (c) the benefit of assigning 60 tasks to 30 nodes; (d) the benefit of assigning 120 tasks to 50 nodes.

Figure 6(a)–(d) visually show the simulation results of DEAEBOA, PSO, GA, BOA, and BWO with 8, 16, 30, and 50 sensor nodes and 15, 30, 60, and 120 tasks, respectively. Overall, DEAEBOA has an improved optimized performance compared to BOA, PSO, GA, and BWO in 4 different situations. Figure 6(a)–(d) show that DEAEBOA has begun to converge by the time the algorithm runs to the 200th generation, at which point DEAEBOA's network benefit are much greater than PSO, GA, BOA, and BWO. At the same time, DEAEBOA shows more obvious advantages in the case of the increasing number of tasks and nodes, and DEAEBOA's benefit also increases more than BOA, PSO, GA, and BWO. It is clear that BOA, PSO, GA, and BWO exhibit a locally optimal state after 100 iterations. Due to the creative addition of adaptive strategies, elite operators, and differential evolution, DEAEBOA is easier to jump out of the local optimal and obtain a better benefit value. Overall, DEAEBOA can deal with the issue of information collection task allocation in SMWSNs when the conditions are the same.

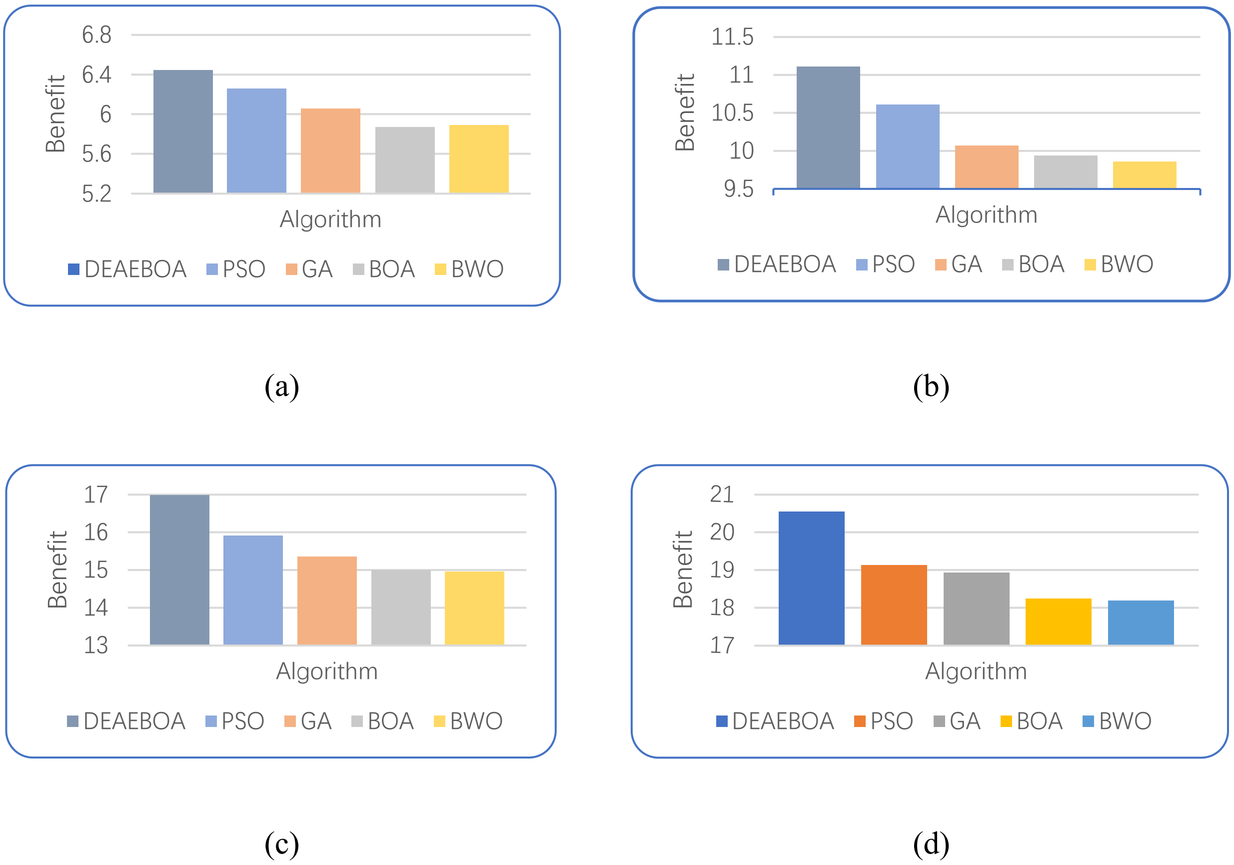

Figure 7. Histogram of the network benefit of the five algorithms under different tasks with 8 nodes: (a) benefit of 15 tasks; (b) benefit of 25 tasks; (c) benefit of 40 tasks; (d) benefit of 50 tasks.

The histogram in Figure 7 shows that when the sensor nodes are 8, the number of tasks is 15, 25, 40, and 50, respectively. The network benefit gap between the five algorithms can be more visualized using the histogram view, and Figure 7(a)–(d) are the result of 150 iterations. It is obvious that DEAEBOA consistently performs the best under these four conditions. When the quantity of nodes is constant, a change in the quantity of tasks will make it continue to grow, the network income becomes higher and higher, and DEAEBOA always performs the best. In Figure 7(d), the network benefit of DEAEBOA is much larger than that of BOA, PSO, GA, and BWO. For agricultural production problems, the information collection effect and economic value can be effectively improved.

Table 4 shows the network benefit for DEAEBOA, PSO, GA, BOA, and BWO with different numbers of nodes and tasks.

Figure 8 shows the benefit growth percentages of the five algorithms corresponding to different task numbers when the number of nodes is 8, 16, 30, and 50, respectively. Data are taken from Table 4, and Figures 6 and 7. As can be seen in Figure 8, DEAEBOA's network benefit growth percentage is greater than that of BOA, PSO, GA, and BWO for seven different tasks. It can be seen that the innovative design of adaptive operators, elite strategies, and differential evolution strategies can jump out of the local optimum, so as to achieve higher network returns and improve the task allocation efficiency of SMWSNs.

Table 4. The network benefit of the five algorithms.

Number of nodes and tasks	DEAEBOA	PSO	GA	BOA	BWO
15 tasks and 8 nodes	6.4452	6.2580	6.0569	5.8687	5.8879
25 tasks and 8 nodes	11.1075	10.6126	10.0674	9.9411	9.8579
40 tasks and 8 nodes	16.9841	15.9036	15.3488	14.9900	14.9503
50 tasks and 8 nodes	20.5443	19.136	18.9283	18.2450	18.1966
30 tasks and 16 nodes	12.6718	11.9088	11.6171	11.2362	11.1878
60 tasks and 30 nodes	23.3405	22.2220	22.0922	21.1183	21.6495
120 tasks and 50 nodes	45.1898	42.3056	42.1824	40.7369	41.4927

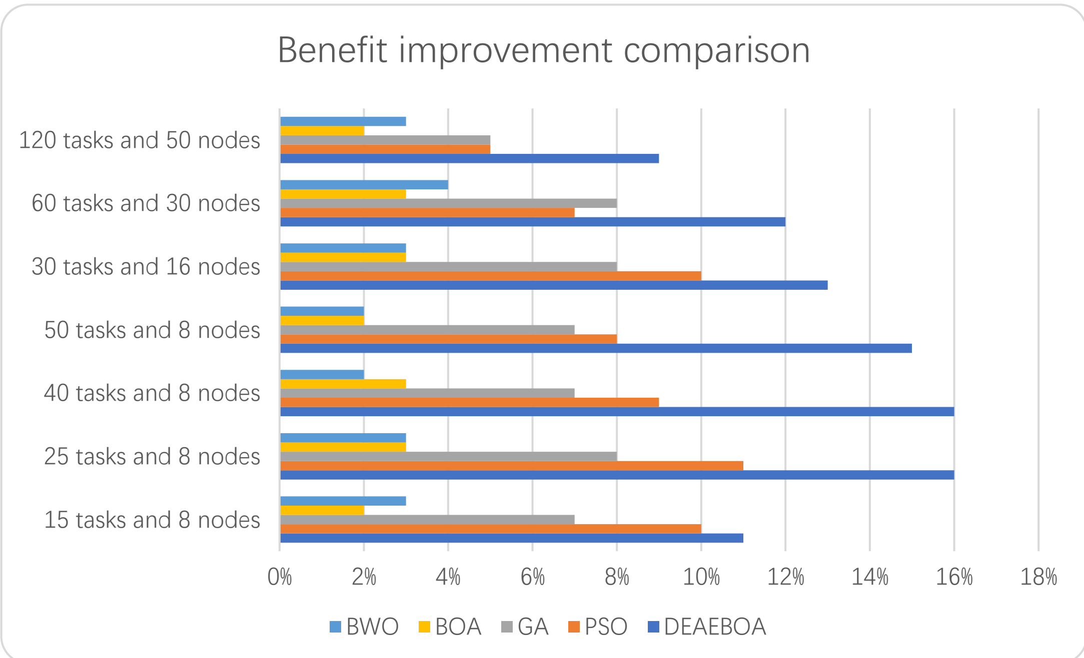

Figure 8. The comparison of the network benefit improvement of the five algorithms.

Figure 9 shows a line chart of the network benefit of the growth of the five algorithms when the quantity of nodes is 8, as the quantity of tasks augments. When the number of tasks is 25 and 40, the network benefit growth rate is highest in DEAEBOA, up to 16%.

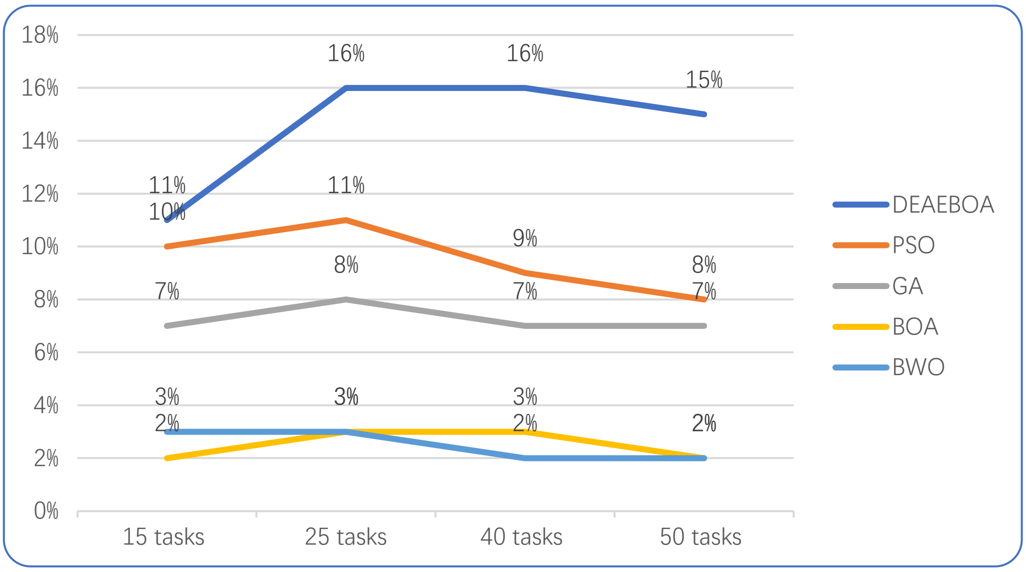

Figure 9. Line chart of the percentage increase in network benefits for the same number of nodes.

Figure 10 displays a line chart of the network benefit growth of the five algorithms under different task numbers and node numbers. In the case shown in Figure 10, the benefit growth percentage of DEAEBOA is invariably greater than that of BOA, PSO, GA, and BWO. In summary, DEAEBOA is very available in resolving the task assignment problem.

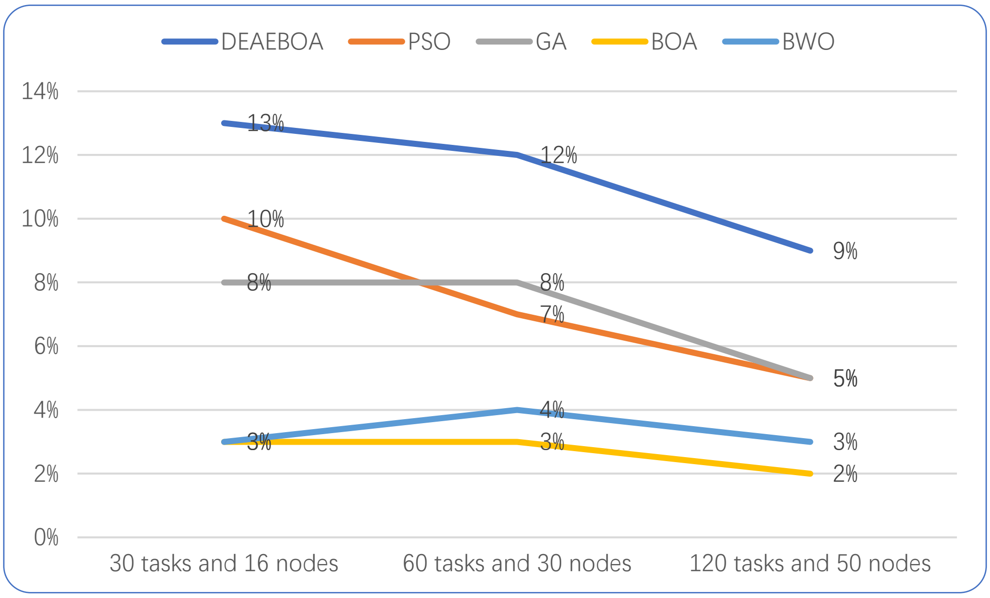

Figure 10. Line chart of network benefit improvement percentage for five algorithms under different number of nodes and number of tasks.

Figure 11 shows the percentage increase in the benefit of DEAEBOA compared to BOA, PSO, GA, and BWO. We can draw according to the data generated under the number of nodes and tasks in Table 4. Obviously, in either case, DEAEBOA is always the best choice for SMWSNs.

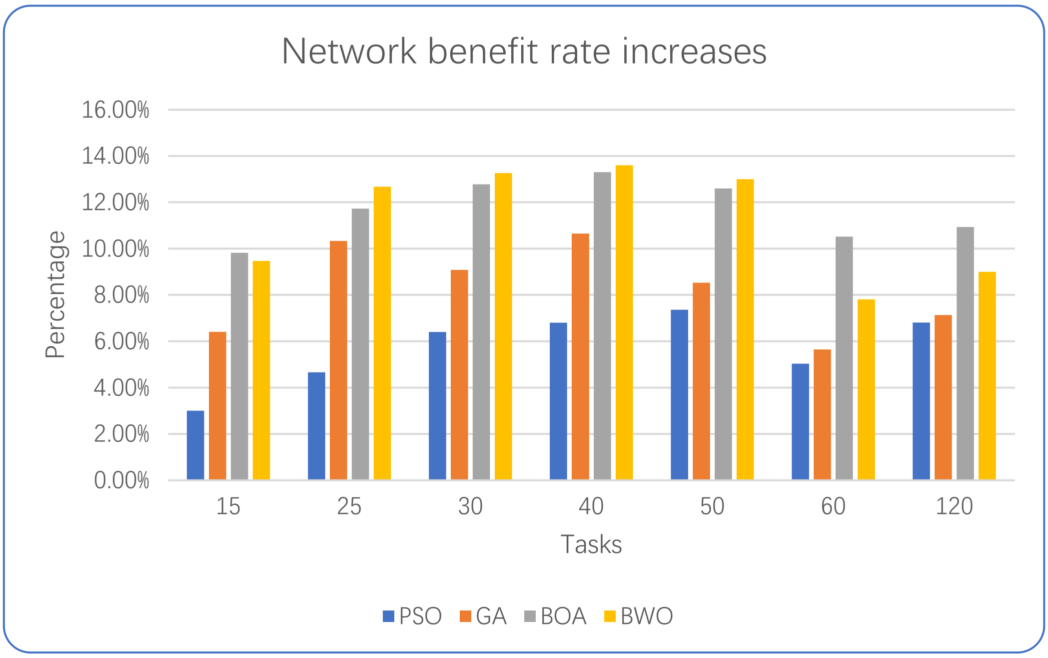

Figure 11. The benefit growth percentage.

Table 4 shows the network benefit improvement rate of DEAEBOA compared to PSO, GA, BOA, and BWO.

Table 5. The network benefit improvement rate.

Number of nodes and tasks	PSO	GA	BOA	BWO
15 tasks and 8 nodes	3.00%	6.41%	9.82%	9.47%
25 tasks and 8 nodes	4.66%	10.33%	11.73%	12.67%
30 tasks and 16 nodes	6.40%	9.08%	12.78%	13.26%
40 tasks and 8 nodes	6.80%	10.65%	13.30%	13.60%
50 tasks and 8 nodes	7.36%	8.53%	12.60%	13.00%
60 tasks and 30 nodes	5.03%	5.65%	10.52%	7.81%
120 tasks and 50 nodes	6.81%	7.13%	10.93%	9.00%

Figure 12 shows the average network benefit improvement rate of DEAEBOA compared to the other four algorithms when the quantity of nodes is 8 and the quantity of tasks is 15, 25, 40, and 50, respectively.

After analyzing and discussing the above simulation results, it proves the effectiveness of the designed DEAEBOA. The new DEAEBOA proposed in this paper has obvious advantages in solving the task allocation problem of SMWSNs, has the advantages of fast running speed and strong optimization ability, and provides reliable help for agricultural production.

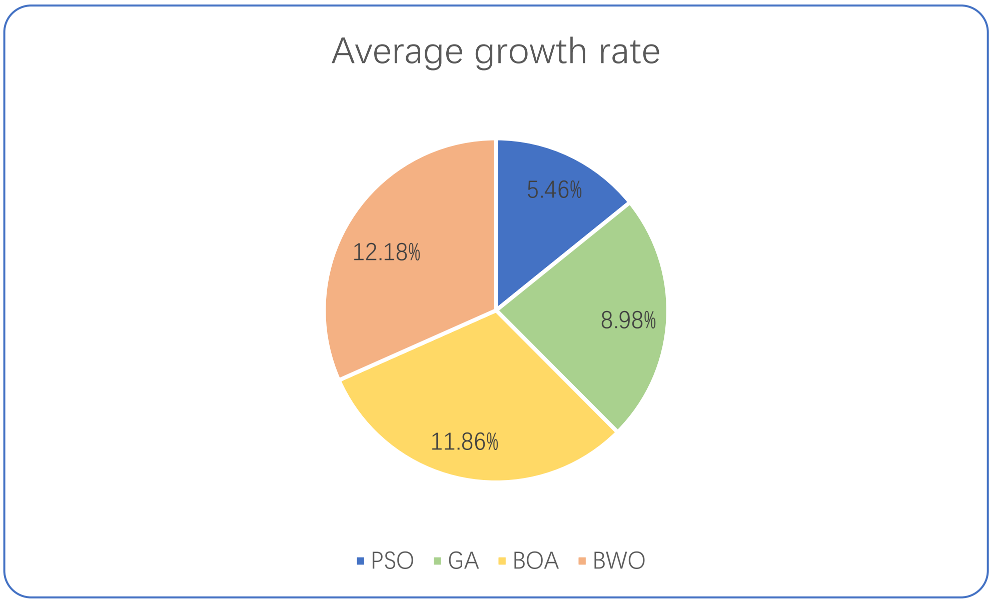

Figure 12. The average network benefit improvement rate.

6. Conclusions and future work

In order to availably improve the efficiency of task allocation, for this purpose, a novel differential evolutionary adaptive elite butterfly optimization algorithm (DEAEBOA) for reliable task allocation of SMWSNs is proposed. Furthermore, a new adaptive strategy and a fresh elite operator are designed, which can enhance the global search capability and avoid falling into local optimization. Not only that, DEAEBOA's network benefit growth rate can reach up to 16% compared to the other four algorithms. Moreover, the network benefit of DEAEBOA has obvious advantages over BOA, PSO, GA, and BWO, and the network benefit rate is increased to 11.86%, 5.46%, 8.98%, and 12.18% respectively. It can be seen that the proposed DEAEBOA can effectively improve the network benefit and performance of SMWSNs; at the same time, this provides a research basis for the further development of intelligent agriculture.

Although the DEAEBOA proposed in this paper has achieved good performance through simulation experiments, proving its superiority, it still needs some improvements due to various constraints, such as the constraints of research capabilities and environmental conditions. In this paper, the sensor nodes of SMWSNs are statically and randomly distributed in the farmland monitoring area. However, with the development of precision agriculture, some application scenarios require the dynamic distribution of sensor nodes, that is, continuous and regular movement to collect and monitor farmland data. In the future, the influence of environmental factors such as temperature, noise, and obstacles will be considered, the dynamic distribution of sensor nodes will be studied, and the network will be placed in three-dimensional space so that the algorithm will be more practical and adaptable.

Use of AI tools declaration

The authors declare they have not used Artificial Intelligence (AI) tools in the creation of this article.

Acknowledgments

This paper was funded by the National Natural Science Foundation of China, grant number 61962053, the Shihezi University High-level Talent Research Start-up Fund Project, grant number RCZK2018C39, the project of Youth and middleaged Scientific and Technological Innovation Leading Talents Program of the Corps, grant number 2018CB006, the Corps innovative talents plan, grant number 2020CB001, the China Postdoctoral Science Foundation, grant number 220531, Funding Project for High Level Talents Research in Shihezi University, grant number RCZK2018C38, Project of Shihezi University, grant number ZZZC201915B, National Key R&D Program of China, grant number 2022ZD0115803, Science and Technology Innovation 2030–"New Generation of Artificial Intelligence" Major Project, grant number 2022ZD0115803.

Conflict of interest

The authors declare no competing interests.

References

1. S. A. Aldalahmeh, D. Ciuonzo, Distributed detection fusion in clustered sensor networks over multiple access fading channels, *IEEE Trans. Signal Inf. Process. Networks*, **8** (2022), 317–329. https://doi.org/10.1109/tsipn.2022.3161827
2. H. Yu, Y. B. Zikria, Cognitive radio networks for internet of things and wireless sensor networks, *Sensors*, **20** (2020). https://doi.org/10.3390/s20185288
3. A. Mohammadi, D. Ciuonzo, A. Khazaee, P. S. Rossi, Generalized locally most powerful tests for distributed sparse signal detection, *IEEE Trans. Signal Inf. Process. Networks*, **8** (2022), 528–542. https://doi.org/10.1109/tsipn.2022.3180682
4. G. N. Kar, P. Verma, S. Mahato, A. Santra, S. Kundu, A. Bose, An IoT-enabled multi-sensor system with location detection for agricultural applications, *Mapan*, **38** (2023), 375–382. https://doi.org/10.1007/s12647-022-00617-7
5. S. K. Sah Tyagi, A. Mukherjee, S. R. Pokhrel, K. K. Hiran, An intelligent and optimal resource allocation approach in sensor networks for smart Agri-IoT, *IEEE Sens. J.*, **21** (2021), 17439–17446. https://doi.org/10.1109/jsen.2020.3020889
6. X. Feng, F. Yan, X. Liu, Study of wireless communication technologies on internet of things for precision agriculture, *Wireless Pers. Commun.*, **108** (2019), 1785–1802. https://doi.org/10.1007/s11277-019-06496-7
7. K. R. Gsangaya, S. S. H. Hajjaj, M. T. H. Sultan, L. S. Hua, Portable, wireless, and effective internet of things-based sensors for precision agriculture, *Int. J. Environ. Sci. Technol.*, **17** (2020), 3901–3916. https://doi.org/10.1007/s13762-020-02737-6
8. L. Yu, W. Gao, R. R. Shamshiri, S. Tao, Y. Ren, Y. Zhang, et al., Review of research progress on soil moisture sensor technology, *Int. J. Agric. Biol. Eng.*, **14** (2021), 32–42. https://doi.org/10.25165/j.ijabe.20211404.6404
9. G. Patrizi, A. Bartolini, L. Ciani, V. Gallo, P. Sommella, M. Carratu, A virtual soil moisture sensor for smart farming using deep learning, *IEEE Trans. Instrum. Meas.*, **71** (2022), 1–11. https://doi.org/10.1109/tim.2022.3196446

10. S. Sakthivel, V. Vivekanandhan, M. Manikandan, Automated irrigation system using improved fuzzy neural network in wireless sensor networks, *Intell. Autom. Soft Comput.*, **35** (2023), 853–866. https://doi.org/10.32604/iasc.2023.026289
11. Z. Cheng, S. Yu, Z. Zhu, L. Zhang, Z. Zhang, Z. Leng, et al., Improvement of defects in soil moisture monitoring of wireless sensor network by mobile sensor platform, in *2018 11th International Conference on Intelligent Computation Technology and Automation (ICICTA)*, (2018), 156–160. https://doi.org/10.1109/ICICTA.2018.00043
12. K. S. Patle, V. Panchal, R. Saini, Y. Agrawal, V. S. Palaparthy, Temperature compensated and soil density calibrated soil moisture profiling sensor with multi-sensing point for in-situ agriculture application, *Measurement*, **201** (2022). https://doi.org/10.1016/j.measurement.2022.111703
13. B. Kashyap, R. Kumar, Sensing methodologies in agriculture for soil moisture and nutrient monitoring, *IEEE Access*, **9** (2021), 14095–14121. https://doi.org/10.1109/access.2021.3052478
14. L. Wang, M. Li, J. Kou, K. Yang, C. Jiang, Adaptive auction protocol for task assignment in wireless sensor and actuator networks, *Int. J. Distrib. Sens. Netw.*, **16** (2020). https://doi.org/10.1177/1550147720932751
15. X. Zhu, K. C. Li, J. Zhang, S. Zhang, Distributed reliable and efficient transmission task assignment for WSNs, *Sensors*, **19** (2019). https://doi.org/10.3390/s19225028
16. X. Liang, S. Li, J. Fei, Adaptive fuzzy global fast terminal sliding mode control for microgyroscope system, *IEEE Access*, **4** (2016), 9681–9688. https://doi.org/10.1109/access.2016.2636901
17. S. Vinod Chandra, H. S. Anand, Nature inspired meta heuristic algorithms for optimization problems, *Computing*, **104** (2021), 251–269. https://doi.org/10.1007/s00607-021-00955-5
18. T. Issac, S. Silas, E. B. Rajsingh, Investigations on PSO based task assignment algorithms for heterogeneous wireless sensor network, in *2019 2nd International Conference on Signal Processing and Communication (ICSPC)*, (2019), 89–93.
19. S. Famila, A. Jawahar, A. Sariga, K. Shankar, Improved artificial bee colony optimization based clustering algorithm for SMART sensor environments, *Peer-to-Peer Networking Appl.*, **13** (2020), 1071–1079. https://doi.org/10.1007/s12083-019-00805-4
20. Y. Hou, Y. Chen, Z. Wang, H. Xiang, Research on dynamic assignment of satellite communication tasks based on GA algorithm, in *IEEE 5th Information Technology and Mechatronics Engineering Conference (ITOEC)*, (2020), 1052–1055. https://doi.org/10.1109/ITOEC49072.2020.9141823
21. D. Shin, A. Kirmani, V. K. Goyal, J. H. Shapiro, Photon-efficient computational 3-D and reflectivity imaging with single-photon detectors, *IEEE Trans. Comput. Imaging*, **1** (2015), 112–125. https://doi.org/10.1109/tci.2015.2453093
22. Y. Gao, R. Ma, H. Cao, C. Yu, G. Ma, H. Xia, et al., Research on task allocation of ground-air collaborative cluster based on two improved firefly algorithms, in *2021 40th Chinese Control Conference (CCC)*, (2021), 1709–1714. https://doi.org/10.23919/CCC52363.2021.9550682
23. Y. Y. Chen, D. Zhang, K. P. Zhu, R. Q. Yan, An adaptive activation transfer learning approach for fault diagnosis, *Ieee-Asme Trans. Mechatron.*, **2023** (2023), 1–12. https://doi.org/10.1109/tmech.2023.3243533
24. W. C. Yeh, Y. Z. Jiang, C. L. Huang, N. N. Xiong, C. F. Hu, Y. H. Yeh, Improve energy consumption and signal transmission quality of routings in wireless sensor networks, *IEEE Access*, **8** (2020), 198254–198264. https://doi.org/10.1109/access.2020.3030629

25. V. Ramsurrun, P. Katsina, S. Anantwar, A. Seeam, S. Cassim, A transmission power optimisation algorithm for wireless sensor networks, in *Towards New e-Infrastructure and e-Services for Developing Countries*, Springer, (2021), 74–85. https://doi.org/10.1007/978-3-030-70572-5_5
26. Z. Hu, L. Xu, L. Cao, S. Liu, Z. Luo, J. Wang, et al., Application of non-orthogonal multiple access in wireless sensor networks for smart agriculture, *IEEE Access*, **7** (2019), 87582–87592. https://doi.org/10.1109/access.2019.2924917
27. J. Xu, Z. Zhang, Z. Hu, L. Du, X. Cai, A many-objective optimized task allocation scheduling model in cloud computing, *Appl. Intell.*, **51** (2020), 3293–3310. https://doi.org/10.1007/s10489-020-01887-x
28. P. Arabas, Modeling and simulation of hierarchical task allocation system for energy-aware HPC clouds, *Simul. Modell. Pract. Theory*, **107** (2021). https://doi.org/10.1016/j.simpat.2020.102221
29. Y. D. Zangue, R. Melot, P. Martin, Diversity of farmland management practices (FMP) and their nexus to environment: A review, *J. Environ. Manage.*, **302** (2022), 114059. https://doi.org/10.1016/j.jenvman.2021.114059
30. A. Srivastava, P. Mishra, A survey on WSN issues with its heuristics and meta-heuristics solutions, *Wireless Pers. Commun.*, **121** (2021), 745–814. https://doi.org/10.1007/s11277-021-08659-x
31. W. Li, S. Zhang, G. Wu, A. Saad, A. Tolba, G. Kim, A sustainable WSN system with heuristic schemes in IIoT, *Comput. Mater. Continua*, **72** (2022), 4215–4231. https://doi.org/10.32604/cmc.2022.024204
32. M. Okhovvat, M. Kheirabadi, A. Nodehi, M. Okhovvat, Task allocation approach for minimizing make-span in wireless sensor actor networks, *Comput. Syst. Sci. Eng.*, **39** (2021), 165–178. https://doi.org/10.32604/csse.2021.05468
33. V. Raee, A. Ebrahimzadeh, R. H. Glitho, H. Elbiaze, Ensuring energy efficiency when dynamically assigning tasks in virtualized wireless sensor networks, *IEEE Trans. Green Commun. Networking*, **6** (2022), 613–628. https://doi.org/10.1109/tgcn.2021.3118967
34. G. S. Kori, M. S. Kakkasageri, Classification and regression tree (CART) based resource allocation scheme for wireless sensor networks, *Comput. Commun.*, **197** (2023), 242–254. https://doi.org/10.1016/j.comcom.2022.11.003
35. H. Baniabdelghany, R. Obermaisser, A. Khalifeh, Reliable task allocation for time-triggered IoT-WSN using discrete particle swarm optimization, *IEEE Internet Things J.*, **9** (2022), 11974–11992. https://doi.org/10.1109/jiot.2021.3132452
36. Z. Zha, C. Li, J. Xiao, Y. Zhang, H. Qin, Y. Liu, et al., An improved adaptive clone genetic algorithm for task allocation optimization in ITWSNs, *J. Sens.*, **2021** (2021), 1–12. https://doi.org/10.1155/2021/5582646
37. M. Xu, J. Zhou, Elite immune ant colony optimization-based task allocation for maximizing task execution efficiency in agricultural wireless sensor networks, *J. Sens.*, **2020** (2020), 1–9. https://doi.org/10.1155/2020/3231864
38. A. Niccolai, F. Grimaccia, M. Mussetta, R. Zich, Optimal task allocation in wireless sensor networks by means of social network optimization, *Mathematics*, **7** (2019). https://doi.org/10.3390/math7040315
39. D. Weikert, C. Steup, D. Atienza, S. Mostaghim, Mobility-aware multi-objective task allocation for wireless sensor networks, in *2021 IEEE Symposium Series on Computational Intelligence (SSCI)*, (2021), 1–8. https://doi.org/10.1109/SSCI50451.2021.9660109

40. S. Arora, S. Singh, Node localization in wireless sensor networks using butterfly optimization algorithm, *Arabian J. Sci. Eng.*, **42** (2017), 3325–3335. https://doi.org/10.1007/s13369-017-2471-9