Architecture designs of dendritic neuron model and swarm intelligence

by

Cheng Tang

A dissertation
submitted to the Graduate School of Science and Engineering
in Partial Fulfillment of the Requirements
for the Degree of
Doctor of Engineering

University of Toyama
Gofuku 3190, Toyama-shi, Toyama 930-8555 Japan

2021
(Submitted October 19, 2021)

Acknowledgements

I deeply appreciate all those who have given me appropriate advice and helpful assistance during my study and research process. This thesis could be successfully completed due to their concern and help.

First of all, I am grateful to Professor Zheng Tang for giving me a glimpse of the charm of artificial intelligence. Thanks for his guidance and help in the dendritic neuron model. With his support and constant encouragement, I could have obtained this degree.

Then, to all members of our laboratory, especially the seniors who have given me a lot of help, for their selfless assistance and patient guidance when I encountered difficulties.

In addition, I have to express my gratitude to the Otsuka Toshimi Scholarship Foundation for providing me with financial support and the confidence to overcome difficulties throughout the entire period of my PhD program.

Last but not least, I would like to thank my family for all of their unconditional love and support for my studies. Their deep love and encouragement accompanied me throughout my entire research, giving me the courage to face setbacks and overcome difficulties.

Abstract

The neural architecture search has gained high importance and effectively improved many machine learning techniques. During my PhD program, I devoted myself to the neural architecture design of dendritic neuron model and swarm intelligence, which are described as follows:

First, dendritic neuron model (DNM), which is a single neuron model with a plastic structure, has been applied to resolve various complicated problems. However, its main learning algorithm, namely the back-propagation (BP) algorithm, suffers from several shortages. That largely limits the performances of the DNM. To address this issue, another bio-inspired learning paradigm, namely the artificial immune system (AIS) is employed to optimize the weights and thresholds of the DNM, which is termed AISDNM. These two methods have advantages on different issues. Due to the powerful global search capability of the AIS, it is considered to be efficient in improving the performance of the DNM. To evaluate the performance of AISDNM, eight classification datasets and eight prediction problems are adopted in our experiments. The experimental results and corresponding statistical analysis confirm the superior performance of the AISDNM when compared with other models. It can be concluded that the reasonable combination of two different bio-inspired learning paradigms is efficient. Furthermore, for the classification problems, empirical evidence also validates the AISDNM can delete superfluous synapses and dendrites to simplify its neural structure, then transform the simplified structure into the logic circuit classifier (LCC). The process does not sacrifice accuracy but significantly improves the classification speed. Based on these results, both the AISDNM and the LCC can be regarded as effective machine learning techniques to solve practical problems.

Second, the scale-free network is well known as an important complex network. The degree of nodes in a scale-free network adheres to a power-law distribution. In the skeleton of the scale-free network, there exists a few nodes which own huge neighborhood size and play a great vital role in information transmission of the entire network, while the majority of the network nodes have few connections whose influences of information exchange are limited to a relatively low level. We introduce a scale-free population topology into the cuckoo search (CS) algorithm to propose a novel variant, which is termed the scale-free cuckoo search (SFCS) algorithm. Unlike other CS algorithms where the individuals exchange information randomly, two properties of a scale-free network can improve the SFCS in two aspects: the possibility that the information of competent individuals quickly floods the whole population is reduced significantly, which guarantees population diversity; and the corrupt individuals can learn from competent individuals with greater probability, which is beneficial for convergence. Thus, SFCS can obtain a better trade-off between exploitation and exploration. To evaluate the effectiveness of the proposed SFCS, 58 benchmark functions with different dimensions (10-Dimension, 30-Dimension, and 50-Dimension), and 21 real-world optimization problems are employed in our experiment. We compare SFCS with the basic CS algorithm, two CS variants, and five state-of-the-art optimization methods, and the corresponding results and statistical analysis verify the superiority of SFCS. Furthermore, SFCS is compared with a scale-free fully informed particle swarm optimization algorithm (SFIPSO) and the experimental results prove our scale-free idea is effective despite its simplicity. We also introduce the scale-free population topology into the differential evolution (DE) and the firefly algorithm (FA) and the additional results show that the scale-free population topology enhance the search ability of the DE and FA. These results lead us to believe that our scale-free population architecture design may be a new perspectives for improving the performance of the population-based algorithms.

The rest of my thesis is structured as follows: first of all, Chapter 1 presents a detailed introduction to the dendritic neuron model and cuckoo search. Chapter 2 reviews the conventional dendritic neuron model and basic cuckoo search algorithm, together with the neural mechanisms and scale-free network. Next, Chapter 3 describes the proposed evolutionary DNM and SFCS algorithm in details. The performance of the AISDNM is evaluated in Chapter 4. Chapter 5 provides an investigation of the SFCS algorithm. Chapter 6 summarizes the conclusion and presents some future research.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Dendritic neuron model

Artificial neural network (ANN) is well-known as one of the respective computational models that inspired by biological neural networks and has recently been applied to diverse engineering and computer science fields [1]. McCulloch and Pitts have first mathematically pioneered the elemental concepts of ANNs [2]. Due to the development of neurobiology and biophysics, the importance of dendritic neural structures in neural computing has been emphasized [3, 4]. Based on the theoretical development of the nerve membrane models and the detailed body of quantitative electrophysiological information, Rall has started the development of mathematical models of dendritic neurons in 1962 [5]. Subsequently, Rall and Rinzel et al. have conducted various researches on a single branch of a dendritic neuron model [6, 7, 8, 9]. Moreover, [10] suggests that single neurons are capable of performing memory, learning, and other specialized cognitive functions in particular dendritic structures.

With the advancement of neuroscience, a δ-like cell model with dendritic morphology has been proposed by Koch, the model analyzes the interactions between excitatory and inhibitory inputs in neural cells [11, 12]. It is confirmed that the model plays an essential role in the retinal ganglion cells [13] and the human auditory system [14]. However, since Koch's model has failed to make any changes on the dendritic structure due to the lack of effective pruning mechanism, it is considered to be implausible in the view of biological neural models [13]. Legenstein and Maass have designed a comprehensive method for nonlinear dendritic calculation, based on synaptic plasticity and branch-strength potentiation. They also provided a mathematical proof that, the synaptic and dendritic plasticity mechanisms can promote rivalry among dendrites, and a individual neuron can perform complex nonlinear functions through appropriate plasticity mechanisms in the dendritic structure [15]. In addition, it has been proven that the evolutionary neural architecture has a strong influence on the performance of ANN [16, 17].

In our previous research, we also proposed a biologically plausible neuron model, which can use a novel dendritic plasticity mechanism to implement different nonlinear functions on dendrites [18]. And a generalized delta-rule-like algorithm is proposed to train its parameters. Further, we have proposed a novel dendritic neuron model (DNM) by modifying the activation functions [19]. The DNM can generate a distinct dendritic neuron morphology for any particular assignment. And its simplified structure allows for the realization in hardware. Since no floating-point computation is in the logic circuit, so the DNM can respond extremely quickly. The neural model has been deployed effectively to solve various complex tasks, such as computer-aided diagnosis [20, 21, 22], transmission trend of the COVID-19 [23], PM2.5 concentration prediction [24] and financial time series prediction [25].

Since the back-propagation (BP) algorithm and its variants have become popular approaches to train ANNs [26], they are also used as the main learning algorithms of the DNM. However, since the BP algorithm utilizes gradient descent to optimize the error function, it is compulsory to use differentiable transfer functions in ANNs. Besides, the BP algorithm also suffers from the following disadvantages, such as the high sensitivity to the initial conditions, slowness of convergence, tendencys to fall into local minimum, and over-fitting problem [27, 28]. Therefore, the BP algorithm has greatly limited the performance of the DNM.

To address these issues, we consider using meta-heuristic algorithms to improve the performance of the DNM. Due to the inspiration from the immune system in vivo, the artificial immune system (AIS) has been widely regarded as an excellent information processing and learnable system, which bridges the research field of immunology and computer science [29, 30, 31, 32]. Because of its powerful search ability, the AIS has achieved considerable success in the field of artificial intelligence [33, 34], and the AIS and its variants have been applied in software personalization [35], classification [36, 37, 38, 39], music piece similarity measures [40] and music recommendation [41]. The AIS mainly includes the following four algorithms. The first is the artificial immune network. It performs immune memory primarily through a reciprocal reinforcement network of B cells [42]. The second is the clonal selection algorithm. It increases population diversity by cloning and hypermutation operators [43]. The negative selection algorithm is the third, which draws inspiration from the negative selection process of T cells [44]. The last is the dendritic cell algorithm, which is derived from the danger theory [45].

Due to distinct advantages, such as few control parameters, simple structure and excellent search ability, the clonal selection algorithms have been regarded as one of the most representative AIS technologies. May et al. have proposed an immune inspired algorithm on the basis of the clonal selection algorithm for the evolution of software test data [46]. Cutello et al. have introduced two exceptional variation operations, namely hypermutation and hypermacromutation, and proposed a novel immune incentive operator into the clonal selection algorithm. The improved variant achieves excellent performance in protein structure prediction problems [47]. Wilson et al. have applied the clonal selection algorithm to solve time series prediction problems [48]. Moreover, the method of hybridizing the clonal selection algorithms with the ANNs was proposed to attack various challenging problems. Jie et al. have designed a multi-user detection technique, in which the Hopfield neural network is employed as the "immune operator" to further improve the affinity of antibodies in the clonal selection algorithm [49]. And the results show that the embedded Hopfield neural network effectively addresses the computational complexity of the clonal selection algorithm and improves the convergence speed. Wang et al. have also introduced the Hopfield neural network into the clonal selection algorithm to further solve the problem of multiple-input multiple-output multiuser detection [50]. Similarly, the techniques to train neural networks by using clone selection algorithms have also obtained impressive results. To achieve the optimal hidden nodes in the cascade–correlation network, Gao et al. have used differential evolution to improve the affinity of the clones of the antibodies and applied the novel clonal selection algorithm to the construction of neural networks [51]. Chitsaz et al. have proposed a new wind power prediction engine on the basis of the wavelet neural network, in which the clonal selection algorithm is utilized to train the forecasting engine. The fusion has been proved to be beneficial to the adjustment of the free parameters of the wavelet neural network [52]. Since the clonal selection algorithm guarantees population diversity and is theoretically able to utilize local characteristic information to prevent the population from being trapped into the local minimum [53], it is considered suitable for improving the computation capacity of the DNM.

In this paper, we leverage recent researches on the clonal selection algorithm to optimize the DNM, by utilizing it as the training algorithm. The major contributions of this research are listed as follows: taking into account the drawbacks of the BP algorithm and the superiority of the AIS algorithm, especially the powerful global search capability, we introduced the AIS into the DNM. The performances of the AISDNM are examined on eight classification datasets and eight prediction problems, compared with other six techniques. The results suggest that the reasonable combination of two different bio-inspired learning paradigms is better than other methods on all datasets. Moreover, when compared with other traditional classifiers, the AISDNM can prune the redundant synaptic layers and useless dendritic layers, thus allowing for simplification of the evolutionary neural structure. The simplified unique topology can be replaced by a logic circuit classifier (LCC). Since the LCC avoids floating-point operations, it can solve complex classification problems with very little computing resources and has almost no effect on accuracy. It can be concluded that the AISDNM and LCC are promising machine learning techniques in the era of big data.

1.2 A cuckoo search algorithm with scale-free population topology

The cuckoo search algorithm, proposed by Xin-She Yang et al. [54], is an efficient and powerful nature-inspired metaheuristic algorithm that addresses optimization issues [55, 56, 57]. The CS algorithm is verified to be capable of converging to the global best solution generally due to the employed Lévy flights. Specifically, the local and global search in the CS are restrained by the switching/discovery selection scheme, which allows the CS algorithm to explore the solution space more efficiently when compared with algorithms implemented by standard random walks [58]. Moreover, the CS algorithm has fewer control parameters to be tuned compared with other metaheuristic algorithms. Therefore, the CS has witnessed rapid developments and has been efficiently applied in numerous fields over the past decade, such as engineering optimization [59], load forecasting [60], surface roughness [61], flow shop scheduling [62], the travelling salesman problem [63] and reliability optimization problems [64]. In addition, various variations of the CS algorithm have been proposed to hasten the convergence and prevent being trapped into local minima in the search process, which can be primarily grouped into three categories.

The first one is hybridization. Li and Yin hybridized the CS algorithm with Nawaz-Enscore-Ham (NEH), which can efficiently generate an initialized population with a specific diversity [62]. In [65], a method of hybridizing the CS with the power series was proposed to solve the electrostatic deflection of micro fixed-fixed actuators. Khan and Sahai combined an ANN with the CS algorithm to assess the performance of computer-aided workstations [66]. Lian et al. combined the CS with the evolutionary strategy in the PSO named PSOCS to solve the optimization problems [67]. Moreover, a hybridization of the krill herd method and the CS algorithm was designed for global optimization tasks [68].

The second category usually embeds newly generated operators. The chaotic operators with a novel strategy of the step size was employed to enhance the search capability of the CS in [69]. Ouaarab et al. incorporated the discrete search mechanism into CS to address the traveling salesman problem [63]. In [70], Walton et al. proposed a modified gradient-free optimization CS model named MCS which increases the information exchange among the top solutions. In addition, Layeb introduced the quantum-inspired computing into the CS algorithm, which contains the superposition of all potential solutions and three novel operations inspired from quantum computing, namely measurement, mutation, and interference [71].

The third category is adopting the adaptive parameter strategy to control the parameters of the CS algorithm. Tuba et al. proposed another modified CS model, where the step size is defined by the sorted function rather than a simple random walk [72]. And Naik and Panda developed a novel variant where the step size of each cuckoo is adapted by its fitness and current position [73]. In [74], the CS algorithm was modified to include a linear decreasing probability mechanism and an adaptive parameter method that increases the diversity of the population.

In addition, the CS has been transformed into a multi-objective optimization algorithm due to its effectiveness and simplicity. Distinct from the single-objective optimization, the multi-objective optimization contains several objectives which contradict each other. Since numerous real-world optimization tasks are generally multi-objective, the multi-objective CS is applied to deal with these complex and highly nonlinear problems, such as design optimization [75], multi-objective unit commitment problem [76] and Jiles-Atherton vector hysteresis parameters estimation [77].

However, most of these studies ignore individual differences in the search process. Recent research has verified that reasonable population structures can significantly enhance the performance of evolutionary algorithms [78, 79, 80]. Thus, numerous evolutionary algorithms have modified the structural topologies to improve their performances, such as the genetic algorithm (GA) [81, 82], PSO [83, 84] and DE [85, 86]. Following this point of view, to introduce a suitable population topology into the CS, we focus on the complex networks which simulate several real-world phenomena, such as space systems, food webs, and collaborative networks. Complex networks consist of classical random networks, small-world networks, and scale-free networks. It is noting that the scale-free networks are considered to be highly appropriate to reconstruct the population topology of the CS algorithm. It is because that, most vertices in scale-free networks are low-degree nodes. Hence, they can effectively control the impact of vertices on the entire network. Additionally, a few nodes with many connections structure the framework, and they are significant roles in the information transmission of the whole network. The scale-free population topology enables the CS algorithm to obtain a better compromise between exploitation and exploration.

These appealing properties suggest that the scale-free network is effective to improve population-based optimization algorithms. Thus the introduction of scale-free networks into evolutionary algorithms has attracted significant attention [87, 88]. Giacobini et al. first introduced the evolutionary algorithms whose populations are constructed in accordance with a scale-free network. Nevertheless, the high selection force induced by scale-free topology leads to premature convergence, and the performance is not superior to the standard panmictic setting [89]. Subsequently, Zhang and Yi designed a novel PSO variant where the Barabási and Albert (BA) scheme was adopted to construct the scale-free population structure [90]. The modified PSO algorithm was verified to improve the performance in dealing with real power loss minimization task [91]. However, the computational complexity is drastically increased in this algorithm because the construction of the population topology is gradually carried out during the optimization process. Compared with the basic algorithm, the improved variant will undoubtedly suffer from a more significant computational burden when solving the same problem.

The main motivation of this research is summarized as follows: first, to use the scale-free network to enables the SFCS to obtain a better agreement between exploitation and exploration and second, to propose a novel scale-free population topology technique for enhancing the search ability of the population-based algorithms. To settle this issue, we novelly introduce the scale-free population topology into the CS algorithm in an efficient way, which is termed the SFCS algorithm. In SFCS, the effect of competent individuals on the whole population is controlled, which ensures the population diversity. While corrupt individuals have a higher probability of learning from competent individuals without paying the cost of random trial and error, which is beneficial for convergence. The computational complexity of the SFCS architecture design is analyzed to verify its computational efficiency. And exhaustive experiments are carried out to evaluate the performances of the SFCS on the benchmark problems, in comparison with the conventional CS, two CS variants, and five metaheuristic optimization algorithms. In addition, the results of parameter sensibility and the performance on real-world tasks are also presented in our study. Finally, we also compare the SFCS with the SFIPSO and introduce the scale-free population topology into the DE and FA. The contributions are generalized as follows: first, a novel mechanism that constructs a scale-free population topology for the CS algorithm is proposed and second, the principle of a scale-free population topology to enhance the search ability of CS is analyzed in this paper and third extensive experimental results verify that the SFCS obtains superior performance than other algorithms. Last but not least, we prove that the SFCS outperforms the SFIPSO where the scale-free network is introduced into the PSO in another way and our scale-free architecture design is capable of improving the performance of SFCS and valid for other population-based algorithms.

Chapter 2

Related works

2.1 Dendritic neuron model

As shown in Fig. 2.1, the structure of the DNM mainly contains the synaptic layer, the dendritic layer, the membrane layer and the cell body.

2.1.1 Synaptic layer

First, input signals of other neurons are delivered to the synaptic layers. In the synaptic layer, the computation performed on these signals can be illustrated as follows:

$$Y_{i,m} = \frac{1}{1 + e^{-k(w_{i,m}x_i - q_{i,m})}}, \tag{2.1}$$

where x_i is the i^{th} input feature. $Y_{i,m}$ denotes the output of the i^{th} synaptic layer on the m^{th} dendrite. k is a user-defined constant parameter. $w_{i,m}$ and $q_{i,m}$ are the connection weight and bias, respectively. Depending on the values of $w_{i,m}$ and $q_{i,m}$, the $\theta_{i,m}$ of each synapse can be determined by:

$$\theta_{i,m} = \frac{q_{i,m}}{w_{i,m}}. \tag{2.2}$$

2.1.2 Dendritic layer

Then, the outputs of the synaptic layers are transmitted to each dendritic layer. The multiplication operation is considered to be an important operation in the nervous

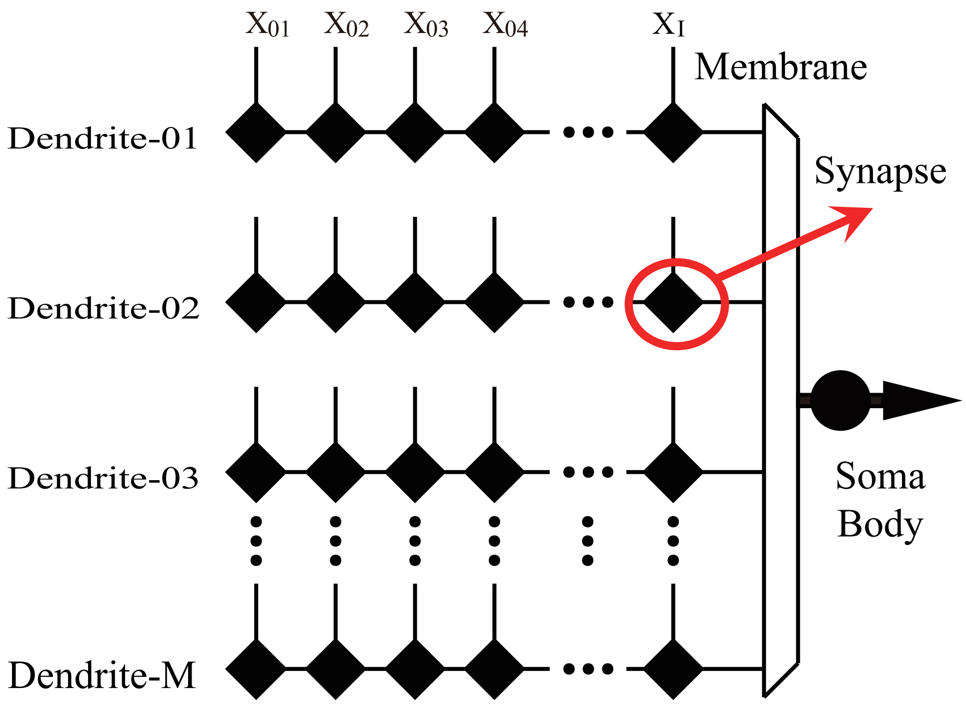

Figure 2.1: The structure topology of the DNM.

system for processing visual [92] and auditory information [93]. Inspired by these biological phenomena, a simplest nonlinear computation named the multiplication is applied to the dendritic layer, which can be defined as follows:

$$Z_m = \prod_{i=1}^{I} Y_{i,m}, \tag{2.3}$$

where Z_m represents the output of the m^{th} dendrite.

2.1.3 Membrane layer

All the results of the dendritic branches are collected and transmitted into membrane layer. A summation operation is used to describe this process, which is formulated by:

$$V = \sum_{m=1}^{M} Z_m, \tag{2.4}$$

where V denotes the output of the membrane layer.

2.1.4 Cell body (Soma)

Finally, the cell body obtains the result of the membrane layer and compares it with its threshold. If the signal strength exceeds the value of threshold, the cell body will fire. Otherwise, it will not fire. Depended on the membrane potential, the state of the cell body is given as follows:

$$O = \frac{1}{1 + e^{-k(V - \theta_{soma})}}, \tag{2.5}$$

where O represents the final neural signal of DNM and θ_{soma} is a user-defined parameter.

2.2 Neural mechanisms

2.2.1 Connection definition

As mentioned in Section 2.1, $w_{i,m}$ and $q_{i,m}$ are modified by the optimization algorithms. Depending on the different combinations of $w_{i,m}$ and $q_{i,m}$, the evolutional directions of synapes are divided into four types, which are illustrated in Fig. 2.2. For a better understand, the mathematical description of each connection type is depicted in Fig. 2.3. From Fig. 2.3, we can observe that, in the synaptic layer of the direct connection, if x_i is larger than $\theta_{i,m}$, the output $Y_{i,m}$ is 1. Otherwise, it will be 0. On the contrary, the synaptic layer of inverse connection implies that, if x_i exceeds $\theta_{i,m}$, its signal will be 0; otherwise, the signal is 1. For the constant 1 connection, the signal $Y_{i,m}$ will maintain at 1 approximately. On the contrary, the synaptic layer will ignore the value of x_i and consistently output 0 for the constant 0 connection. By the definition of distinct synaptic layers, the DNM can perform a unique pruning mechanism to simply its neural structure.

2.2.2 Synaptic pruning

The synaptic pruning mechanism can remove the redundant synaptic layers in DNM. As introduced above, the output of the synaptic layer in the constant 1 connection case is always 1. Since a multiplication operation is performed in the dendrite, the synaptic layer has no effect on the results of the dendrite, according to the rule 'any value multiplied by 1 is equal to itself'. As shown in Fig. 2.4, the synapses in the constant 1 connection case can be deleted completely.

2.2.3 Dendritic pruning

The dendritic pruning mechanism can discard unnecessary dendritic layers. Similarly, the output of the synaptic layer in the constant 0 connection case is 0. Based on the rule 'any value multiplied by 0 is equal to 0', the signal of the whole dendritic layer is 0. In other words, these dendritic layer cannot contribute to the result of the membrane layer. Thus, DNM needs to remove this kind of dendritic layers in DNM, which has been shown in Fig. 2.4.

2.2.4 Hardware implementation

Through the synaptic and dendritic pruning, DNM can generate a unique and simplified structure for each specific task. Furthermore, the simplified structure can be replaced by an LCC. For example, as shown in Fig. 2.5, the function of the synapses in the direct connection case is replaced by a comparator. While the function of the synapses in the inverse connection case can be realized by a comparator and a logic NOT gate. The dendritic layer actually implements a logical conjunction function, which can be approximately substituted by a logic AND gate. The function of the membrane layer is nearly a logical disjunction function, which can be implemented by a logic OR gate. Finally, the soma body can simply be replaced by a wire. In this way, an LCC can be obtained to approximate the function of the DNM. It is easy for hardware implementation. All the computation of the LCC is a binary operation, rather than the floating-point operation of the DNM and other ANNs. It can vastly improve the computation speed of the DNM.

2.3 Cuckoo search algorithm (CS)

Cuckoos is an exotic kind of birds because of their pleasant sounds and particular breeding strategies, for instance, they are parasitic in that they lay eggs in other birds' nests (generally other species). They even remove the eggs of their hosts in order to maximize the probability of incubation of their eggs [94]. Besides, it is found that fruit flies or Drosophila melanogaster may suddenly turn 90 degrees in their flight direction while searching for food, which is called Lévy flights [95]. Numerous researches have suggested that the movement patterns of various species show the ordinary feature of Lévy flights [96, 97]. Inspired by the nest parasite of cuckoos and the Lévy flights, the CS was designed. The procedure of the CS is employed by the following principles:

(1) Every cuckoo lays a egg in each iteration, and parasitizes a random host's nest;

(2) The nests that have the highest qualified eggs (solutions) are retained in the offspring generation;

(3) The host can identify a parasitic egg by using a certain probability ($P_a \in [0, 1]$)in a fixed number of host nests.

Initially, each host nest is randomly assigned to an egg, which is given as follows:

$$x_{i,j} = L_j + rand(0, 1)(U_j - L_j), \tag{2.6}$$

where L_j and U_j denote the prescribed minimum and maximum boundaries, respectively, of the j^{th} dimensional variable. $i \in [1, 2, ..., N]$ and N is the overall number of host nests. $j \in [1, 2, ..., D]$, and D denotes the dimensional number.

Next, cuckoos explore and exploit the new nests, and the CS algorithm combines local random search and global exploratory random search in a balanced manner, which is controlled by the parameter P_a. The local random search can be expressed as follows:

$$X_i^{t+1} = X_i^t + \alpha * s \oplus H(P_a - \varepsilon) \oplus (X_i^t - X_k^t), \tag{2.7}$$

where X_i^{t+1} is the new nest searched by the i-th cuckoo in the $t+1^{th}$ iteration. X_i^t and X_k^t are two different solutions in the t^{th} iteration. s represents the step size, α indicates the scaling factor of s, which is a user-defined parameter. ε is a random element with uniform distribution. $H(u)$ denotes a Heaviside function. The Lévy flights are adopted to perform the global exploratory random search which can be determined by:

$$X_i^{t+1} = X_i^t + \alpha * L(s, \lambda), \tag{2.8}$$

where

$$L(s, \lambda) = \frac{\lambda \Gamma(\lambda) \sin(\pi\lambda/2)}{\pi} \frac{1}{s^{1+\lambda}}, 1 < \lambda < 3. \tag{2.9}$$

Eq. (2.9) is a stochastic equation. Since the next solution only depends on the conversion probability and the current solution, random search of the CS can be generally regarded as a type of Markov chain. Then, a small portion of the worst nests are discarded and the new nests are established. According to the principles mentioned above, the pseudocode of the main process of the CS algorithm is provided in Algorithm 1.

Algorithm 1 The pseudocode of the CS algorithm

```
Initialize the host nests and evaluate the fitness of each nest;
repeat
    Seek the new nests via Lévy flights and evaluate the fitness of each new nest;
    Select an old nest randomly and compare it with the new one;
    Determine whether to accept the new nest or not, according to the greedy selection mechanism;
    Abandon a small portion of the worst nests, and build the new nests via Lévy flights;
    Rank the nests via their fitness and find the current best;
until (The stopping condition is met.)
```

2.4 Scale-free network

The degree of nodes is exponentially decreasing in this kind of network. This property has been found in the majority of real-world biological networks [98]. In general, the distribution of node degree in the scale-free network can be expressed by:

$$P(k) \propto k^{-\alpha}, \tag{2.10}$$

in which $P(k)$ denotes the anticipation that a random node has a degree k, and it is proportional to $(1/k)^{\alpha}$. α represents the scaling exponent, and it ranges in $[2, 3]$ for most of the real-world scale-free models. As shown in Fig. 2.6, we provide the degree distribution of scale-free models in two different ways. The common method is displaying the distribution of degrees using a straight scale for the k and $P(k)$ axes. While the same degree distribution are presented on a logarithmic scale for the k and $P(k)$ axes, a straight line can be observed obviously. Barabási and Albert first designed the scheme to establish such a scale-free model, namely BA algorithm. In their procedure, the growth of the model is combined by attaching new nodes to existing nodes with specific preferences. The main procedure of the BA algorithm is also provided.

(1) Initially, the scale-free architecture begins from a simple structure of M_0 completely connected nodes;

(2) Calculate the probability by: $p(u) = \delta(u)/\sum_j \delta(j)$, and $\delta(u)$ denotes the linking degree of u^{th} node ;

(3) Add a new vertex and attach it to an existing node according to the preference $p(u)$;

(4) Updated the degree of all the nodes;

(5) The previous Step 2-4 are repeated until $(N - M_0)$ nodes have been added to the model.

Barabási et al. demonstrated that the connection probability $P(k)$ of the model established by the BA algorithm is proportional to k^{-3}. Therefore, the distribution of vertex degrees in this model is considered to follow the power law distribution. Fig. 2.7 illustrates a scale-free model built by the BA algorithm and a random model.

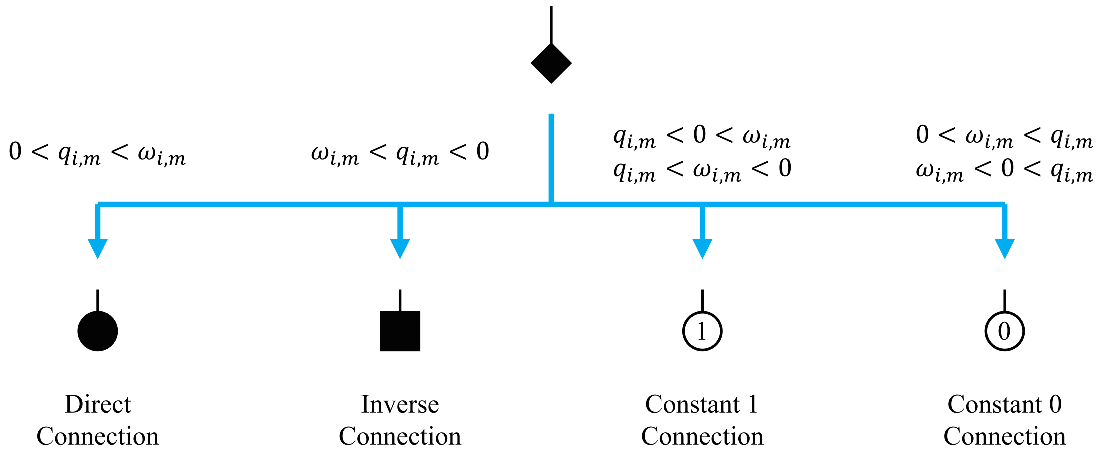

Figure 2.2: Four connection cases of the synaptic layers.

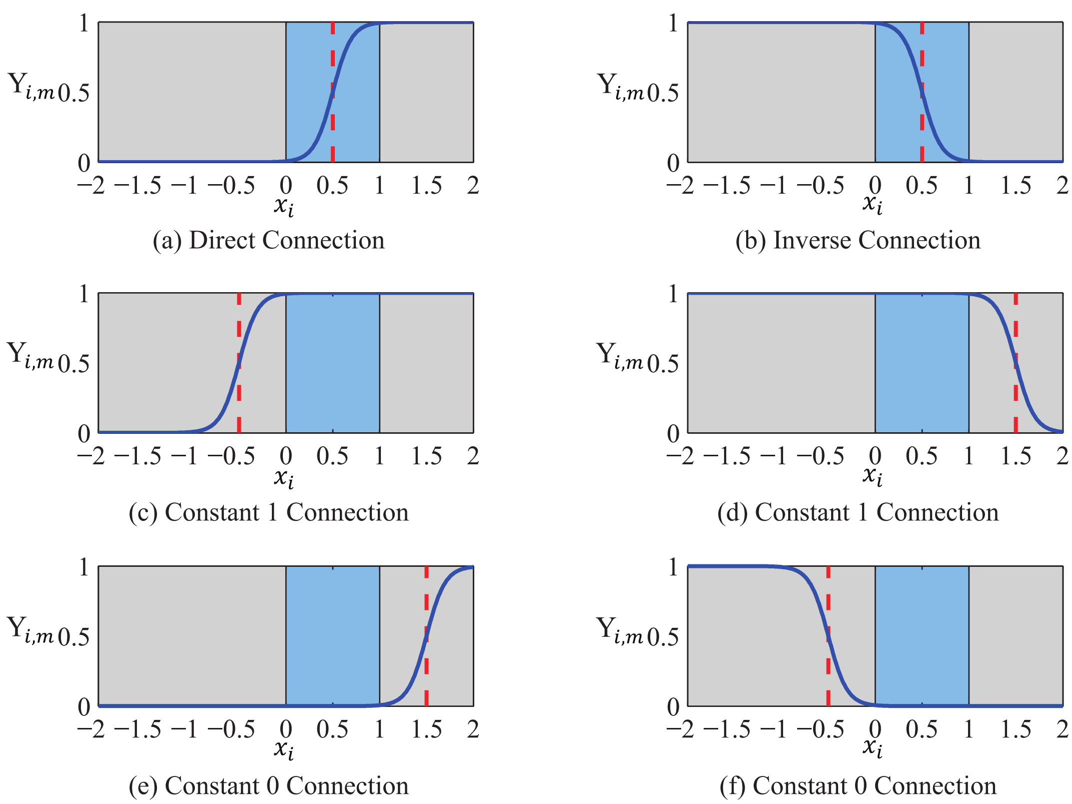

Figure 2.3: Six types of parameter settings.

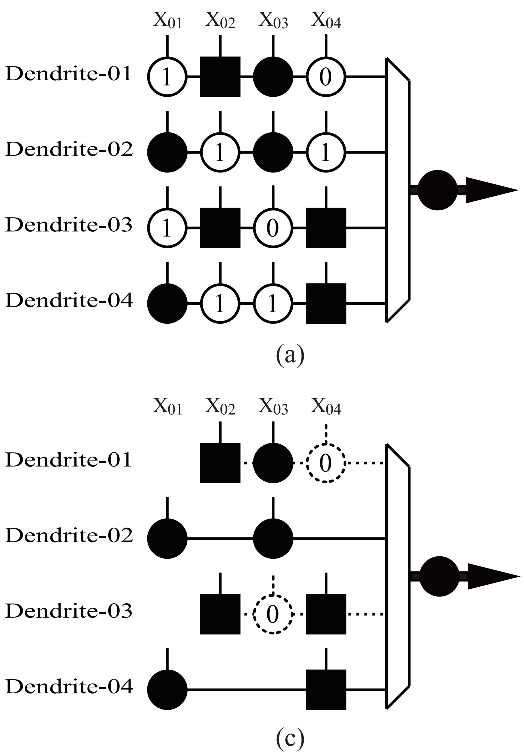

Figure 2.4: Synaptic and dendritic pruning.

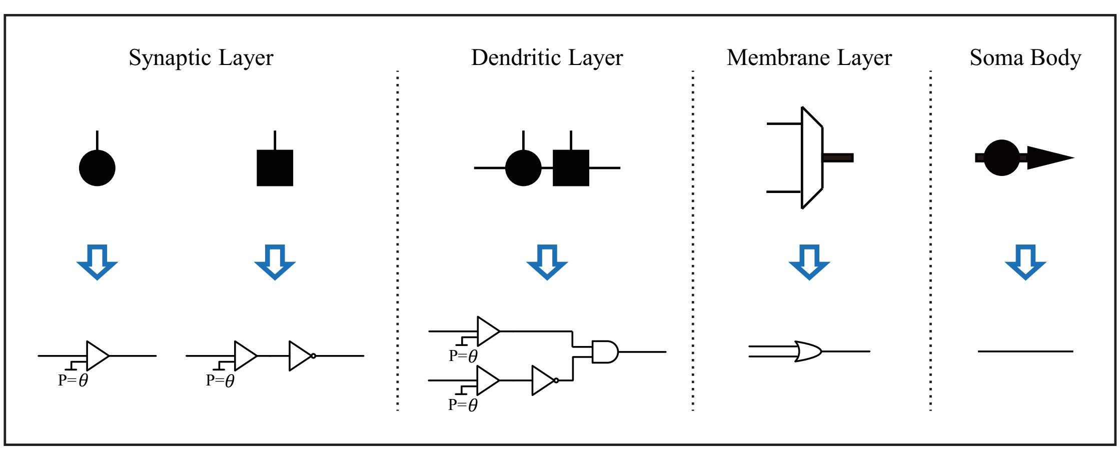

Figure 2.5: The logic circuit simulation of the DNM.

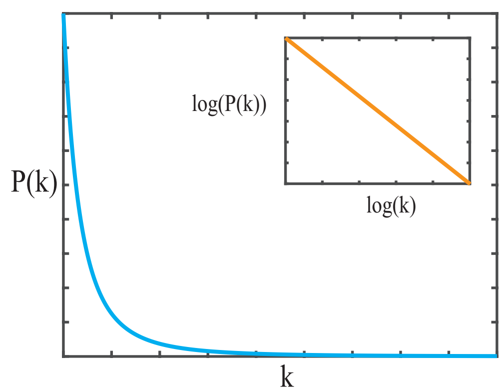

Figure 2.6: Degree distribution of a scale-free network.

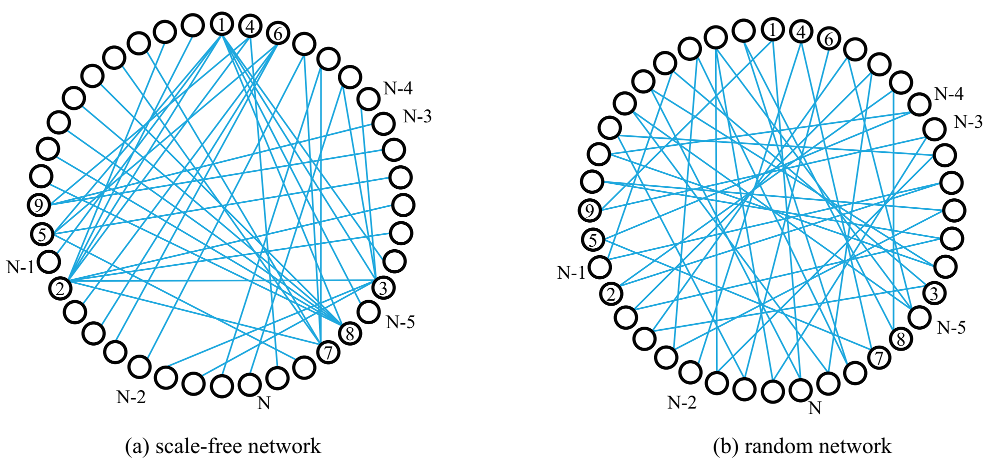

Figure 2.7: Structural topologies of a scale-free architecture and a random architecture.

Chapter 3

Method

3.1 Artificial immune system (AIS)

The AIS has been regarded as a promising metaheuristic algorithm [99]. Compared with other metaheuristic algorithms, the AIS has a stronger capability of preventing the population from falling into the local minimum. The local information can be utilized to improve global parallel computing and suppress repetitive and futile work in the process, which makes crossovers and mutations more efficient. Inspired by the immune phenomenon in vivo, degradation is also used in the evolutionary process of the population in the AIS, which enables the population to increase steadily and healthily [100]. The purpose of introducing the AISDNM is to theoretically use the local feature information of the AIS to help the DNM escape from the local minimum. In general, the AIS is mainly implemented through four steps, namely, cloning, hypermutation, crossover and vaccination. Then, the immune selection is employed to prevent deterioration.

In the AIS, the population Pop_A is initialized and each individual is assigned a solution randomly by:

$$x_{i,j} = L_j + rand(0,1)(U_j - L_j), \tag{3.1}$$

where $i \in [1, N]$, N is the number of individuals in the population, and $j \in [1, D]$, D represents the dimensional value of the objective function. L_j and U_j denote the specified upper and lower bounds of the j^{th} dimensional variable. $[x_{i,1}, x_{i,2}, ..., x_{i,D}]$ form the solution X_i of the i^{th} individual. The fitness function of each individual is calculated by the objective function.

Then, the vaccine population Pop_{Va} are selected from the population Pop_A, and the cloning operator is employed to clone them t times to generate the intermediate population Pop_{Clo} [101]. Next, the hypermutation operators are performed on the clone population Pop_{Clo}. In this study, the proportional mutation operation is utilized to enhance the searching ability of the AIS. The crossover operators act on the population Pop_A to produce the population Pop_B. The information validity of the vaccine population extracted from the existing individuals plays a crucial role in accelerating the method to converge. Then, the vaccination operators execute the population Pop_B in a point-to-point manner to generate population Pop_C.

The immune selection contains two steps. The first is the immune test. If the fitness score of the offspring is worse than that of the original individual, which means that degradation has occurred, the parent will be selected to participate in the next iteration. The second is the anneal selection. All the new individuals are selected in a probabilistic manner. The probability can be calculated by:

$$P(x_i) = \frac{e^{fit(x_i)/T_k}}{\sum_{i=1}^{N} e^{fit(x_i)/T_k}}. \tag{3.2}$$

$fit(x_i)$ denotes the fitness value of x_i and T_k denotes a temperature-controlled series. The primary process of the AIS is presented in Algorithm 2. In addition, the flow chart of the AISDNM is described in Fig. 3.1.

3.2 Scale-free cuckoo search algorithm (SFCS)

In the section, the overall skeleton of the SFCS is first given. Then, a detailed description of the tion is presented. Finally, the computational complexity is analyzed.

Algorithm 2 Artificial immune system.

```
Initialize the population Pop_A;
repeat
    Extract the vaccine population Pop_Va from prior knowledge;
    Perform the cloning operator on the vaccine population Pop_Va and obtain the
  population Pop_Clo;
    Perform the hypermutation operator on the population Pop_Clo;
    Perform the crossover operator on the population Pop_A and obtain the pop-
  ulation Pop_B;
    Perform the vaccination operator on the population Pop_B and obtain the
  population Pop_C;
    Perform the immune selection on the population Pop_Clo and Pop_C, and obtain
  the next population Pop_A;
until (the termination requirement is satisfied.)
```

3.2.1 Framework

First, similar to the original CS algorithm, the proposed SFCS algorithm initializes the population consisting of N nests. Second, we establish a scale-free population topology having N nodes according to the BA algorithm, which is presented in Fig. 2.7(a) Third, each nest is allocated with a fitness value regarding the evaluation function of different problems. Fourth, the nests are also sorted in descending order of the nests' fitness values, and all nests correspond one-to-one the nodes in the scale-free population topology. Finally, each solution corrects its position by learning from one of its neighbors $X^{neighbor}$, which are adjacent in a scale-free population topology. The flow chart of the SFCS is presented in Fig. 3.2.

3.2.2 Motivation

From Fig. 3.2, it can be easy to observe that the crucial part of SFCS is motivation, except fitness evaluation and population sorting. The mechanism of motivation is demonstrated in detail in this section. First, all nests are sorted in descending order and are placed into each node of the scale-free population topology in accordance with its label. Consequently, the nest which owns the high fitness locals into the node with the high degree, while the nest which has bad fitness locals into the node with the low degree in the scale-free population topology. In other words, good solutions

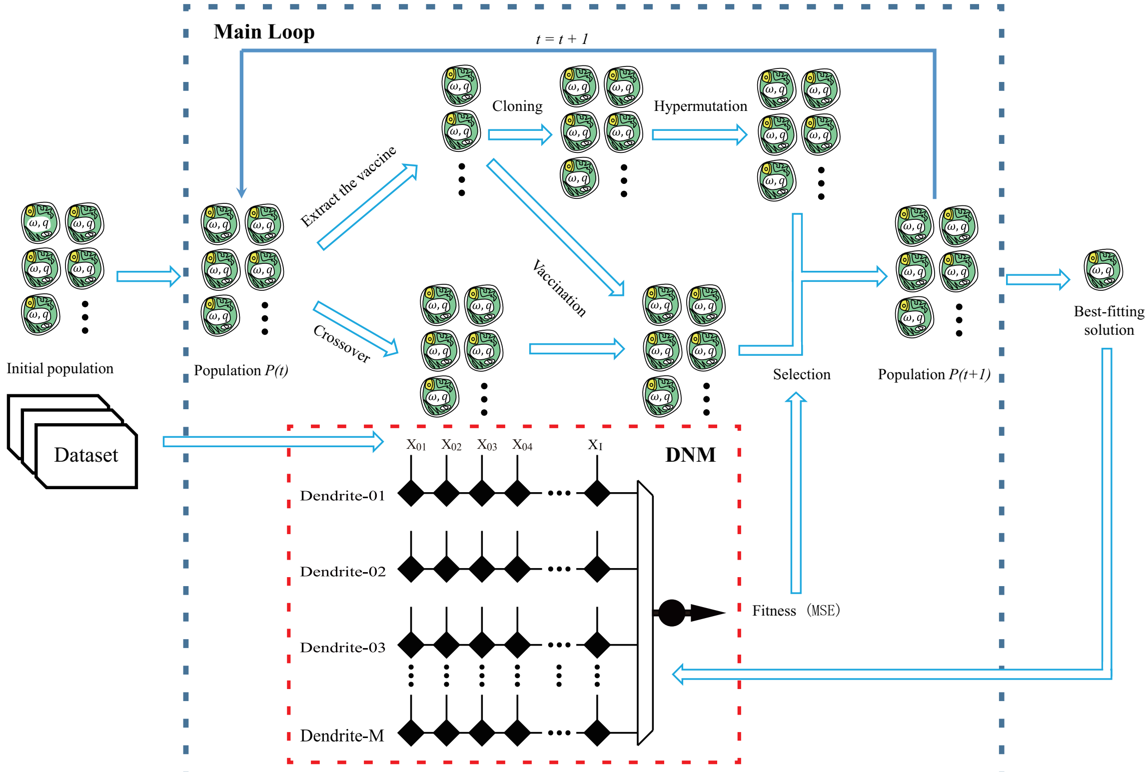

Figure 3.1: Flowchart of the evolutionary neural architecture design methodology.

correspond to more neighbors. but the good ones own fewer neighbors. The power-law distribution characteristic illustrates that there exists a few high-degree nodes link to the majority of nodes in the network, which is obvious to observe in Fig. 2.7. Compared with the random network, good solutions can easily spread their information in the scale-free population topology. Besides, the power-law distribution also ensures there exist numerous nodes, which implies the bad solutions are difficult to impose ineffective information on other nodes. In general, the power-law distribution characteristic is effective to improve the convergence speed of the SFCS.

Additionally, the scale-free network has the other remarkable attribute that it owns a low assortativity. The degree-degree correlation coefficient gauges the extent that the high-degree nodes attach to each other. A low assortativity suggests that the connections between the high-degree nodes are relatively fewer, and the information exchange between the good solutions are less frequent in the scale-free network. It can effectively prevent some good solution rapidly from taking over the entire pop-

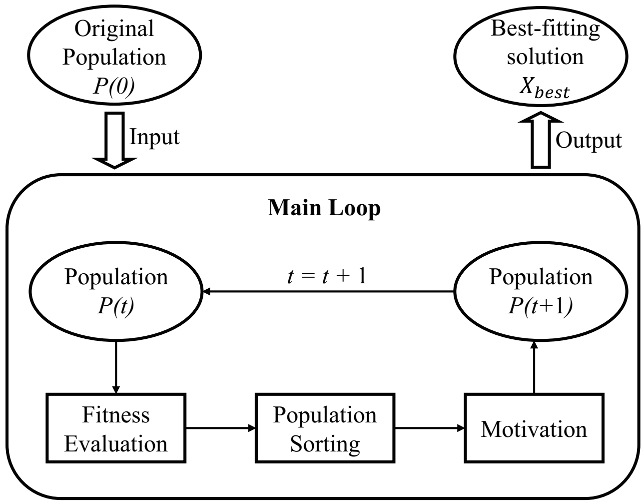

Figure 3.2: Flowchart of the SFCS algorithm.

ulation. Thus, the low degree-degree correlation coefficient attribute is conducive to maintaining the diversity in the scale-free network. It is worth mentioning that all the nests are sorted again in each iteration and we place them in order into the nodes of the scale-free population topology. After that, each nest randomly selects a neighbor nest as the parasitic object to correct the current position. Hence, the new update mechanism of the SFCS algorithm can be expressed by:

$$X_i^{t+1} = X_i^t + \alpha * s \oplus H(P_a - \varepsilon) \oplus (X_i^t - X_{neighbor}^t), \tag{3.3}$$

where $X_{neighbor}^t$ represents a random solution which is selected from the neighbors individuals of X_i^t.

3.2.3 Computational complexity

Finally, a study of the computational complexity is conducted to evaluate the effectiveness of SFCS. T represents largest iteration number, and N indicates the population size of the host nests. As shown in Algorithm 1, under the most unfavorable condition, the complexity of CS can be described as follows:

(1) The complexity of the population initial phase is $O(N)$;

(2) The evaluation of fitness values requires $O(N)$;

(3) The local random search requires $O(N)$;

(4) The computational complexity of performing the global exploratory random search is $O(N)$;

(5) The population sorting costs $N * \log(N)$.

Therefore, the total complexity of the CS algorithm in the worst situation is $O(T * (N * \log(N) + 4N) + 2N)$. The upper boundary of time complexity is worth addressing, so the complexity of the algorithm can be expressed to $O(T * N * \log(N))$. Furthermore, the analysis of the time complexity of the SFCS algorithm is accessible, owing to the clarity of our proposed mechanism. Apart from the principal framework of the algorithm, the additional calculation is the construction of a scale-free population topology, which at most requires $O(N * \log(N))$. Consequently, the overall computational complexity of the SFCS algorithm is $O(T * (N * \log(N) + 4N) + N * \log(N) + 2N)$, which is slightly larger than that of the CS algorithm. However, the simplified result is still $O(T * N * \log(N))$ which is equal to that of CS. Although the SFCS architecture design is slightly inferior to the CS algorithm when the maximum number of iterations is the same, we prove that SFCS architecture design has a more comparable and efficient convergence capability and can obtain a better solution. We can conclude that the computational efficiency of the SFCS architecture design is superior to that of the basic CS algorithm.

Chapter 4

Experimental studies of evolutionary dendritic neuron model

4.1 Experimental setup

The MLP and the conventional DNM are used as the competitors of the AISDNM. For all the three models, the maximum iteration number is set to 1000, and all experiments are conducted 30 times independently. Eight classification datasets and eight prediction problems are adopted in the experiments. 50% of the samples in each dataset are used for training and the rest are used to test the performances of the models. In order to prevent small numeric attributes from being taken over by large numeric attributes, we normalize all values, which can be described as follows:

$$x_{normalized} = \frac{x_{original} - x_{min}}{x_{max} - x_{min}}. \tag{4.1}$$

$x_{normalized}$ represents the normalized data and $x_{original}$ represents the original value. x_{min} and x_{max} denote the maximum and minimum values of x_i in all samples, respectively.

4.2 Datasets description

4.2.1 Classification datasets

Eight classification datasets are employed in the experiments, which include Breast, Glass, Haberman, Iris, Thyroid, Wine, Rice and Heart, which are summarized in Table 4.1. Each dataset can be acquired from the UCI Machine Learning Repository [102]. There are 699 cases in the Breast cancer dataset. The number of features is nine. It is worth noting that only 683 cases are adopted in the experiment because 16 cases have incomplete feature values. All the cases are divided into benign or malignant instances. The Glass dataset includes 214 glass samples. Each instance has nine features. According to these chemical components, the glass samples are classified into the window or the non-window category. The Haberman dataset comprises samples come from a survey of the survival of patients who have undergone breast cancer surgery, which is carried out by the University of Chicago's Billings Hospital. The dataset records three characteristics of 306 samples, which are labeled into two categories according to whether the patient survived within five years after the surgery. Depending on their four attributes, the Iris dataset divides 150 iris instances into Setosa, Versicolor and Virginica types. Among them, two categories are nonlinearly separable from each other, while the latter is divisible from the other two linearly. The Thyroid dataset is supported by the Garavan Institute. It contains 215 samples, which are classified into three categories, and each sample has five characteristics. The Wine dataset is the result of the analysis of wines derived from three different breeds. It consists of 178 instances and each instance has 13 constituents. The Rice dataset is provided by Cinar and Koklu [103]. In this dataset, a total of 3810 rice images are taken, processed and feature inferred. Each grain of rice has seven morphological characteristics. The Heart dataset contains twelve clinical characteristics of 299 patients [104]. These medical information is recorded during their follow-up period and labeled into two categories.

Table 4.1: The detail of eight classification datasets.

Dataset	Num. of classes	Num. of features	Num. of samples
Breast	2	9	683
Glass	2	9	214
Haberman	2	3	306
Iris	2	4	150
Thyroid	2	5	215
Wine	2	13	178
Rice	2	7	3810
Heart	2	12	299

4.2.2 Prediction problems

The prediction problems involve BoxJenkins, EEG, MackeyGlass, Tourism and four chaotic maps, which are listed in Table 4.2. The BoxJenkins times series dataset can be found in [105]. The EEG dataset is provided by Zak Keirn from the Electrical Engineering Department of Purdue University in his Masters of Science thesis. Complete information can refer to `https://www.cs.colostate.edu/eeg/main/data/1989_Keirn_and_Aunon`. The MackeyGlass dataset is produced by a nonlinear time-delay differential equation, which can be expressed as follows:

$$\frac{dx}{dt} = \beta \frac{x_\tau}{1 + x_\tau^n} - \gamma x, \tag{4.2}$$

where β, τ, n and γ represent the real numbers. x_τ is the value of x at $t - \tau$. And the Tourism dataset records the number of monthly forecast tourists arrival to Japan, which can be downloaded from `https://statistics.jnto.go.jp`. The first chaotic map is a typical logistic map. The iterator represents the chaotic behavior in a dynamic system, which can be described as follows:

$$y_{i+1} = 4y_i(1 - y_i), \tag{4.3}$$

where y_i denotes the i^{th} value in the map, and y_0 is equal to 0.152. The second chaotic map is a piecewise linear chaotic map, which is an invariant density function

Table 4.2: The detail of eight prediction datasets.

Dataset	Num. of instance
BoxJenkins	292
EEG	2492
MackeyGlass	981
Tourism	138
Chaos-01	475
Chaos-02	469
Chaos-03	469
Chaos-04	472

in a defined interval. It is determined by:

$$y_{i+1} = \begin{cases} \frac{y_i}{0.7}, & y_i \in (0, 0.7] \\ 0.3(1 - y_i), & y_i \in (0.7, 1) \end{cases}, \tag{4.4}$$

where y_0 is set to 0.002. The singer map is adopted as the third chaotic map and given as follows:

$$y_{i+1} = 1.073(7.86y_i - 23.31y_i^2 + 28.75y_i^3 - 13.302875y_i^4), \tag{4.5}$$

where y_0 is equal to 0.152. The final chaotic map is the sine map. It is generated by the following equation:

$$y_{i+1} = \sin\left(\pi y_i\right), \tag{4.6}$$

where y_0 is set to 0.152.

4.3 Performance evaluation criteria

For the classification datasets, a number of performance evaluation criteria are utilized to estimate all the three models. Specifically, the accuracy on 30 independently runs are used as the four evaluation metrics. Besides, the nonparametric statistical analysis, named the Wilcoxon signed-rank test, is also employed to distinguish whether there exists a significant difference between the AISDNM and its competitor [106, 107]. The null hypothesis suggests no significant difference. When applying a statistical procedure to reject a hypothesis, a level of significance is used to determine at which level the hypothesis may be rejected. Accordingly, the p value represents the probability of assuming that the null hypothesis is true.

In addition, another five comprehensive performance indicators are also used in the experiments. The true positive cases (TP) is the number of instances whose actual and predicted results are both positive. The true negative cases (TN) represents the number of instances that are classified as negative and the corresponding actual classes are negative as well. The false positive cases (FP) denotes the number of samples that are detected as positive, but the actual categories are negative. The false negative cases (FN) is the number of samples whose predicted categories are negative, while their actual categories are positive. The sensitivity (TPR) represents the capability of the technique to identify positive samples, and it can be described by $Sensitivity$=TP/(TP+FN). The specificity (TNR) evaluates the ability of the classifier to classify negative instances, which is calculated by $Specificity$=TN/(FP+TN) (also called Recall). The false positive rate (FPR) represents the percentage of samples that are detected as positive, but in fact, they are negative. It can be computed by FPR=FP/(FP+TN). The false negative rate (FNR) is the percentage of instances that are classified as negative, while they are positive in actual. It can be defined by FNR=FN/(TP+FN). The common evaluation criterion named $F_{measure}$ is also used in our experiments [108]. It can be defined by:

$$F_{measure} = 2 \times \frac{Precision \times Recall}{Precision + Recall}, \tag{4.7}$$

where $Precision$ is defined by:

$$Precision = \frac{TP}{TP + FP}. \tag{4.8}$$

In addition, a large value of the Cohen's Kappa (K) denotes the better classifier is excellent [109]. It can be described by

$$K = \frac{P_o - P_e}{1 - P_e}, \tag{4.9}$$

where P_o denotes the agreement probability between actual and predicted classification and P_e represents the hypothetical chance consistency probability. The AUC ranges from 0 to 1, and the value of the AUC closing to 1 corresponds to a reliable classifier [110]. Last but not least, the convergence speed is the final evaluation metric. It compares the average best-so-far solution of all models in each iteration and the convergence curves are presented.

For the prediction problems, we adopt six performance evaluation criteria to evaluate all the models. The MSE is the first evaluation metric, which can be calculated as follows:

$$MSE = \frac{1}{M}\sum_{m=1}^{M}(O_m - T_m)^2, \tag{4.10}$$

where O_m and T_m are the actual output and target of the m^{th} sample, respectively.

The second evaluation criterion is the mean absolute percentage error (MAPE). It is an important statistical measure of prediction accuracy, which can be calculated as follows:

$$MAPE = \sum_{m=1}^{M}\left|\frac{O_m - T_m}{T_m}\right|. \tag{4.11}$$

The mean absolute error (MAE) is employed as the third evaluation criterion, and it is defined as follows:

$$MAE = \frac{1}{M}\sum_{m=1}^{M}\left|O_m - T_m\right|. \tag{4.12}$$

The fourth evaluation metric is the correlation coefficient (R) [111], which is a famous goodness-of-fit measure for the standard regression model and linear regressions are also provided.

$$R = \frac{\sum_{m=1}^{M}(T_m - \overline{T})(O_m - \overline{O})}{\sqrt{\sum_{m=1}^{M}\left(T_m - \overline{T}\right)^2\left(O_m - \overline{O}\right)^2}}, \tag{4.13}$$

where $\overline{O}$ and $\overline{T}$ represent the mean values of the vectors O and T.

Similarly, the nonparametric statistical analysis and convergence speed are also utilized to compare the performances of the algorithms.

4.3.1 Sensitivity analysis of user-defined parameters

It is well known that the selected values of the parameters have a strong effect on the performance. Various methods have been provided for parameter setting, such as adaptive parameter mechanisms [112, 113, 114, 115, 116], parameter-setting-free mechanisms [117, 118] and random disturbance strategy [119]. The Taguchi method is utilized to analyze the parameter selection of AISDNM [120]. There are three critical user-defined parameters in the AISDNM, namely, the parameter of the connection sigmoid functions in synaptic layer and soma body (k), the number of dendrites (M), and the threshold of the soma body (θ_{soma}). According to the previous studies, there are 4 levels of interest in each parameter, and the full factorial analysis requires the total $4^3 = 64$ experiments [19, 18]. Thus, it will be extremely time-consuming. The Taguchi method can utilize the orthogonal $L_{16}(4^3)$ array to effectively reduce the number of experiments. Only 16 experiments are carried out to achieve a suitable parameter setting of the AISDNM. The best parameter setting of all the classification and prediction problems are presented in Table 4.5 $-$ 4.20.

In addition, for a relatively fair comparison, the number of weights and thresholds are set to be as equal as possible. The number of the adjusted weights in the MLP (N_{MLP}) is calculated as follows:

$$N_{MLP} = I \times L + 2 \times L + 1, \tag{4.14}$$

where I refers to the number of features, and L is hidden layers in the MLP. The adjusted weights number in AISDNM (N_{DNM}) can be determined by:

$$N_{DNM} = 2 \times I \times J, \tag{4.15}$$

Table 4.3: Parameter settings of three models for eight classification datasets.

Datasets	Model	No. of inputs	No. of branches/hidden layers	Learning rate	No. of adjusted weights
Breast	MLP	9	15	0.01	162
	DNM	9	9	0.01	166
	AISDNM	9	9	-	166
Glass	MLP	9	21	0.01	232
	DNM	9	13	0.01	234
	AISDNM	9	13	-	234
Haberman	MLP	3	6	0.01	31
	DNM	3	5	0.01	30
	AISDNM	3	5	-	30
Iris	MLP	4	13	0.01	79
	DNM	4	10	0.01	80
	AISDNM	4	10	-	80
Thyroid	MLP	5	13	0.01	92
	DNM	5	9	0.01	90
	AISDNM	5	9	-	90
Wine	MLP	13	29	0.01	436
	DNM	13	17	0.01	442
	AISDNM	13	17	-	442
Rice	MLP	7	20	0.01	181
	DNM	7	13	0.01	182
	AISDNM	7	13	-	182
Heart	MLP	12	31	0.01	435
	DNM	12	18	0.01	432
	AISDNM	12	18	-	432

where J denotes the number of dendrites. Table 4.3 and Table 4.4 illustrate the comparison of the number of the adjusted weight on classification datasets and prediction problems, respectively.

4.4 Comparison of the classification datasets

In the classification datasets, the AISDNM achieves better results than the MLP, DT, line-SVM, rbf-SVM, poly-SVM and DNM on most datasets. As shown in Table 4.21, the AISDNM outperforms the MLP, DT, line-SVM and DNM on the Breast dataset in terms of Max, Min, *Average* and *Std*. The rbf-SVM and poly-SVM achieve better results than the AISDNM in terms of Min, but the AISDNM obtains better results in all other evaluation criteria. The p values calculated by the Wilcoxon signed-rank test are inferior than the significant level, indicating that the AISDNM is significantly superior to the MLP, DT, line-SVM and DNM on the Breast dataset. For the Glass dataset, the AISDNM is better than the MLP, DT, line-SVM, rbf-SVM, poly-SVM and DNM in terms of most evaluation criteria. The exception can be found that

Table 4.4: Parameter settings of three models for eight prediction datasets.

Datasets	Model	No. of inputs	No. of branches/hidden layers	Learning rate	No. of adjusted weights
BoxJenkins	MLP	2	6	0.01	25
	DNM	2	6	0.01	24
	AISDNM	2	6	-	24
EEG	MLP	4	13	0.01	79
	DNM	4	10	0.01	80
	AISDNM	4	10	-	80
MackeyGlass	MLP	4	13	0.01	79
	DNM	4	10	0.01	80
	AISDNM	4	10	-	80
Tourism	MLP	6	15	0.01	121
	DNM	6	10	0.01	120
	AISDNM	6	10	-	120
Chaos-01	MLP	4	11	0.01	67
	DNM	4	8	0.01	64
	AISDNM	4	8	-	64
Chaos-02	MLP	4	11	0.01	67
	DNM	4	8	0.01	64
	AISDNM	4	8	-	64
Chaos-03	MLP	4	11	0.01	67
	DNM	4	8	0.01	64
	AISDNM	4	8	-	64
Chaos-04	MLP	4	13	0.01	79
	DNM	4	10	0.01	80
	AISDNM	4	10	-	80

the AISDNM, poly-SVM and DNM obtain the same result of Max accuracy. The statistical results imply that the AISDNM has a satisfactory performance on the Glass dataset, compared with the MLP, DT, line-SVM, rbf-SVM, poly-SVM and DNM. Besides, the AISDNM has the best performance on the Haberman dataset. The MLP and DNM achieve the same results in terms of Min accuracy. The corresponding p values suggest that the AISDNM significantly outperforms the MLP, DT, line-SVM, rbf-SVM, poly-SVM and DNM. Although the MLP, DT, rbf-SVM and poly-SVM obtain the best Max accuracy on the Iris dataset, which is slightly superior to that of the DNM and the AISDNM, the AISDNM achieves the best results in terms of Min, $Average$ and Std, and the statistical results imply that the AISDNM is significantly better than the MLP, DT, line-SVM and DNM on the Iris dataset. Compared with the AISDNM, the DT and DNM have a comparable performance of Max accuracy on the Thyroid dataset. But they cannot perform as well as the AISDNM in terms of the other evaluation criteria. The MLP, line-SVM, rbf-SVM, poly-SVM are inferior to AISDNM in terms of all the evaluation criteria. The corresponding p values imply that the AISDNM has significant better performances than the MLP, DT, line-SVM,

Table 4.5: $L_{16}(4^3)$ The Taguchi's experimental result of the Breast dataset.

No.	k	m	θ_{soma}	Acc(Average±Std)(%)
1	**2**	**9**	**0.2**	**96.49±0.82**
2	2	11	0.4	96.28±0.76
3	2	13	0.6	96.42±0.75
4	2	15	0.8	96.46±0.79
5	5	9	0.4	95.90±0.82
6	5	11	0.2	95.99±0.68
7	5	13	0.8	95.71±0.84
8	5	15	0.6	95.94±0.64
9	8	9	0.6	95.70±0.73
10	8	11	0.8	95.50±0.90
11	8	13	0.2	95.79±1.12
12	8	15	0.4	95.88±0.88
13	10	9	0.8	95.69±1.12
14	10	11	0.6	95.30±1.02
15	10	13	0.4	95.70±0.92
16	10	15	0.2	95.62±0.74

rbf-SVM, poly-SVM and DNM on the Thyroid dataset. For the Wine dataset, the AISDNM achieves a better performance than MLP, DT and DNM in terms of *Average* and *Std*. From the statistical results, it is easy to observe that the AISDNM is significantly superior to the DT and DNM. The AISDNM outperforms the MLP, DT and DNM on the Rice dataset in terms of *Max*, *Min*, *Average* and *Std*. Although the line-SVM, rbf-SVM and poly-SVM achieve better results than the AISDNM in terms of *Min* accuracy, the AISDNM has better results in all other evaluation criteria. The statistical results demonstrate that the AISDNM has a satisfactory performance on the Rice dataset, compared with the MLP, DT and DNM. For the Heart dataset, the AISDNM is better than the MLP, DT, line-SVM, rbf-SVM, poly-SVM and DNM in terms of all evaluation criteria. The corresponding p values imply that the AISDNM has significant better performances than the MLP, DT, rbf-SVM and poly-SVM on the Heart dataset, while there are no significant differences among the AISDNM, line-SVM and DNM.

To further confirm the performance of the AISDNM, the scores of the additional metrics are summarized in Table 4.22. It can be found that, for the Breast datasets,

Table 4.6: $L_{16}(4^3)$ The Taguchi's experimental result of the Glass dataset.

No.	k	m	θ_{soma}	Acc(Average±Std)(%)
1	2	9	0.2	91.93±2.39
2	2	11	0.4	92.83±2.59
3	2	13	0.6	92.96±2.53
4	2	15	0.8	93.40±2.00
5	5	9	0.4	93.36±2.26
6	5	11	0.2	93.30±2.44
7	**5**	**13**	**0.8**	**94.17±1.65**
8	5	15	0.6	93.77±2.29
9	8	9	0.6	92.99±1.87
10	8	11	0.8	93.52±2.05
11	8	13	0.2	93.21±2.45
12	8	15	0.4	93.71±2.16
13	10	9	0.8	93.30±2.06
14	10	11	0.6	93.30±2.19
15	10	13	0.4	93.12±2.23
16	10	15	0.2	92.68±2.50

the AISDNM obtains better performances than the MLP, DT, line-SVM, rbf-SVM, poly-SVM and the DNM in terms of $Sensitivity$, $Specificity$, $F_{measure}$, K and AUC. The AISDNM outperforms the MLP, DT, line-SVM, rbf-SVM, poly-SVM and the DNM in terms of $Sensitivity$, $Specificity$ and K on the Glass dataset. The exceptions can be found that the poly-SVM and MLP have the best performances of $Sensitivity$ and AUC on Glass dataset, respectively. For the Haberman dataset, although the line-SVM and DT perform best in $Sensitivity$ and $Specificity$ respectively, and the DNM has the best performance of AUC, the AISDNM is slightly inferior to them and obtains the best results of $F_{measure}$ and K. For the Iris dataset, the AISDNM has better results than the MLP, DT, line-SVM, rbf-SVM, poly-SVM and the DNM in terms of K and AUC. The AISDNM outperforms the MLP, DT, line-SVM, rbf-SVM, poly-SVM and the DNM in terms of $Specificity$, $F_{measure}$, K and AUC on the Thyroid dataset, and the only exception is that the $Sensitivity$ of the AISDNM is slightly inferior to the MLP, line-SVM, rbf-SVM and poly-SVM. For the Wine dataset, the AISDNM has the best result only in terms of AUC. The poly-SVM and DT obtain the best $Sensitivity$ and AUC respectively on the Rice dataset,

Table 4.7: $L_{16}(4^3)$ The Taguchi's experimental result of the Haberman dataset.

No.	k	m	θ_{soma}	Acc(Average±Std)(%)
1	2	3	0.2	73.62±3.49
2	**2**	**5**	**0.4**	**74.68±2.57**
3	2	7	0.6	74.01±2.55
4	2	9	0.8	74.36±2.56
5	5	3	0.4	72.48±3.53
6	5	5	0.2	73.33±1.84
7	5	7	0.8	72.72±3.40
8	5	9	0.6	73.29±2.70
9	8	3	0.6	73.14±3.46
10	8	5	0.8	73.20±2.88
11	8	7	0.2	73.05±3.20
12	8	9	0.4	73.55±3.75
13	10	3	0.8	72.66±2.72
14	10	5	0.6	73.86±2.79
15	10	7	0.4	73.83±2.75
16	10	9	0.2	72.42±2.71

which are slightly superior to those of the AISDNM, but the AISDNM achieves better performances than the MLP, DT, line-SVM, rbf-SVM, poly-SVM and the DNM in terms of $Specificity$, $F_{measure}$ and AUC. For the Heart dataset, although the DNM has a better result than the AISDNM in terms of $Sensitivity$, the $Specificity$ of the DNM is quite inferior to the AISDNM. The DT obtains the best $Specificity$ which is slightly superior to the AISDNM, but the $Sensitivity$ of the AISDNM is drastically better than that of the DT. The AISDNM outperforms the MLP, DT, line-SVM, rbf-SVM, poly-SVM and the DNM in terms of $F_{measure}$, K and AUC on the Heart dataset. The ROC curves of all the models are shown in Fig. 4.1. It is obvious that the AISDNM achieves larger calculated areas under the ROC curves than the MLP, DT, line-SVM, line-SVM, poly-SVM and DNM for most problems, which implies that the AISDNM has an excellent classification performance. Furthermore, the convergence curves of the MLP, DNM and AISDNM for all datasets are compared in Fig. 4.2. It can be easy to observe that, the convergence speed of AISDNM is more agile than those of the MLP and DNM for all the classification datasets, which confirms that the AISDNM consumes less computing resources when solving the same

Table 4.8: $L_{16}(4^3)$ The Taguchi's experimental result of the Iris dataset.

No.	k	m	θ_{soma}	Acc(Average±Std)(%)
1	2	4	0.2	89.60±4.56
2	2	6	0.4	93.56±3.71
3	2	8	0.6	94.53±3.65
4	2	10	0.8	94.27±2.36
5	5	4	0.4	94.62±1.90
6	5	6	0.2	94.18±1.87
7	5	8	0.8	93.82±2.83
8	**5**	**10**	**0.6**	**95.29±1.28**
9	8	4	0.6	94.04±2.09
10	8	6	0.8	94.09±2.40
11	8	8	0.2	93.78±2.05
12	8	10	0.4	93.42±2.12
13	10	4	0.8	94.40±2.44
14	10	6	0.6	93.42±2.20
15	10	8	0.4	93.69±2.71
16	10	10	0.2	93.82±2.14

problem. In general, based on the above results, the AISDNM can be regarded as an effective classifier in terms of distinct evaluation criteria.

In addition, both the 5-fold cross-validation (CV) and 10-fold CV methods are also performed to estimate the robustness of the AISDNM. It has been proven that repeated cross-validation with different k-fold subsets can only slightly reduce the variance of the estimated performance measures [121]. However, it is worth noting that the training sets of each CV folds contradict the independence assumption of the standard statistical test, which will underestimate the variance of the estimated performance measure [122]. Thus, each CV method is conducted 30 times independently in the experiment. Based on the result of the 30 times repeated CV experiment, the Wilcoxon signed-rank test is also utilized. From Table 4.23, it is easy to observe that, in both the 5-fold and 10-fold CV experiments, the AISDNM has the best results on the Glass, Haberman, Thyroid, Rice and Heart datasets in terms of *Average* and *Std*. The exceptions can be found that, the poly-SVM and rbf-SVM have the best results of *Average* on the Iris and Wine datasets respectively in both the 5-fold and 10-fold CV experiments, but those of the AISDNM are slightly inferior to them.

Table 4.9: $L_{16}(4^3)$ The Taguchi's experimental result of the Thyroid dataset.

No.	k	m	θ_{soma}	Acc(Average±Std)(%)
1	2	5	0.2	85.90±3.53
2	2	7	0.4	87.72±2.88
3	2	9	0.6	89.01±2.74
4	2	11	0.8	89.57±2.80
5	5	5	0.4	92.38±2.00
6	5	7	0.2	92.53±2.68
7	**5**	**9**	**0.8**	**93.98±2.73**
8	5	11	0.6	92.81±2.13
9	8	5	0.6	92.13±2.23
10	8	7	0.8	93.21±1.86
11	8	9	0.2	92.47±2.36
12	8	11	0.4	92.19±2.29
13	10	5	0.8	91.45±2.70
14	10	7	0.6	92.35±2.78
15	10	9	0.4	91.67±2.33
16	10	11	0.2	91.73±2.14

The rbf-SVM and poly-SVM achieve the best performances on the Breast dataset in the 5-fold and 10-fold experiments, respectively. Again, the statistical results imply that the AISDNM is significantly superior to the MLP, DT, line-SVM, rbf-SVM, poly-SVM and DNM on most of the datasets. The exceptions can be found that no significant differences are detected between the AISDNM and the MLP on both the Wine dataset. Compared with the AISDNM, the DT and line-SVM have comparable performances on the Iris and Wine dataset, respectively. The AISDNM performs similar to the rbf-SVM and poly-SVM on the Breast, Iris and Wine datasets. According to the above results, we can conclude that the AISDNM has better robustness than the MLP, DT, line-SVM, rbf-SVM, poly-SVM and DNM.

4.5 Neuronal pruning and hardware implementation

According to the aforementioned neuronal pruning mechanism, the dendritic morphology can be reconstructed for each specified problem. The redundant synaptic

Table 4.10: $L_{16}(4^3)$ The Taguchi's experimental result of the Wine dataset.

No.	k	m	θ_{soma}	Acc(Average±Std)(%)
1	2	13	0.2	91.12±2.82
2	2	15	0.4	94.42±2.14
3	2	17	0.6	94.19±2.34
4	2	19	0.8	94.19±2.32
5	5	13	0.4	94.42±1.82
6	5	15	0.2	94.64±2.26
7	**5**	**17**	**0.8**	**94.98±2.62**
8	5	19	0.6	93.75±2.88
9	8	13	0.6	93.52±2.35
10	8	15	0.8	94.04±2.90
11	8	17	0.2	92.96±2.87
12	8	19	0.4	93.22±2.47
13	10	13	0.8	93.52±3.48
14	10	15	0.6	93.41±2.51
15	10	17	0.4	92.62±3.35
16	10	19	0.2	93.03±2.52

and dendritic layers will be removed. The evolution of the structures of the AISDNM on each classification dataset is provided in Fig. 4.3 − 4.10. It can be found that, the neural structure of the AISDNM is largely simplified compared with the original one, only a few useful synapses and dendritic branches are retained. Moreover, the corresponding LCCs for each classification problem are also presented in Fig. 4.11 − 4.18, and the P represents the threshold $\theta_{i,m}$ of the corresponding synaptic layer. The LCC is only composed of the comparators, the logic AND, OR and NOT gates. The classification accuracies of the LCCs are compared with the AISDNM in Table 4.24. We can observe that the accuracies of the LCCs are almost the same as those of AISDNM. Thus, it can be concluded that replacing the AISDNM with the LCC will not sacrifice accuracy on the classification problems.

4.5.1 Comparison of the prediction problems

The results of prediction problems are summarized in Table 4.25 and 4.26. It is obvious that the AISDNM obtains better results than the MLP, DT, line-SVM, rbf-SVM, poly-SVM and DNM in terms of MSE, MAPE, MAE, and R on most datasets.

Table 4.11: $L_{16}(4^3)$ The Taguchi's experimental result of the Rice dataset.

No.	k	m	θ_{soma}	Acc(Average±Std)(%)
1	2	7	0.2	91.74±0.55
2	2	9	0.4	92.75±0.47
3	2	11	0.6	92.81±0.28
4	**2**	**13**	**0.8**	**92.92±0.46**
5	5	7	0.4	92.55±0.37
6	5	9	0.2	92.68±0.34
7	5	11	0.8	92.82±0.27
8	5	13	0.6	92.83±0.47
9	8	7	0.6	92.58±0.40
10	8	9	0.8	92.81±0.41
11	8	11	0.2	92.68±0.36
12	8	13	0.4	92.81±0.50
13	10	7	0.8	92.54±0.46
14	10	9	0.6	92.80±0.41
15	10	11	0.4	92.75±0.32
16	10	13	0.2	92.73±0.47

The exception can be observe that the DT has the best MAPE on the BoxJenkins, MackeyGlass and Tourism datasets. Besides, the line-SVM obtains the largest R on the EEG dataset. The results of the AISDNM in the rest evaluation criteria on all datasets are all best among the six models. Statistically, the corresponding p values further confirm that the AISDNM has significantly better performances on most of the datasets. In addition, the convergence of the three methods is presented in Fig. 4.19, and the AISDNM has the best performance. For the sake of simplicity, only the fitting graphs and the corresponding linear regression graphs of the AISDNM in the training and prediction phases are plotted in Fig. 4.20 and 4.21. As shown in the left panels, the plot on the left side of the black solid line and that on the right side are the training and prediction phase, respectively. The blue represents the target data and red lines denote the predicted data. The corresponding linear regression graphs and R are plotted in the right-hand set. According to these results, we can conclude that the distributions of points are approximately fitted to the regression lines on most datasets. However, due to the attributes of the DNM, the performances of the AISDNM at the peaks and valleys of the data is not satisfactory, which also can be

Table 4.12: $L_{16}(4^3)$ The Taguchi's experimental result of the Heart dataset.

No.	k	m	θ_{soma}	Acc(Average±Std)(%)
1	2	12	0.2	80.49±3.84
2	2	14	0.4	81.56±3.69
3	2	16	0.6	81.60±2.70
4	**2**	**18**	**0.8**	**82.07±2.80**
5	5	12	0.4	81.09±2.28
6	5	14	0.2	81.96±2.60
7	5	16	0.8	81.58±2.88
8	5	18	0.6	81.04±2.75
9	8	12	0.6	80.67±2.66
10	8	14	0.8	81.20±2.63
11	8	16	0.2	80.00±2.88
12	8	18	0.4	80.29±2.61
13	10	12	0.8	81.04±3.23
14	10	14	0.6	79.84±2.54
15	10	16	0.4	81.04±3.17
16	10	18	0.2	79.73±3.34

observed in the corresponding linear regression graphs. Therefore, there are still some improvements that can be performed to prevent the overestimation of valleys and the underestimation of peaks.

Table 4.13: $L_{16}(4^3)$ The Taguchi's experimental result of the BoxJenkins dataset.

No.	k	m	θ_{soma}	MSE(Average±Std)
1	2	2	0.2	2.75e-02±8.15e-05
2	2	4	0.4	1.44e-02±2.55e-04
3	2	6	0.6	9.83e-03±2.05e-04
4	2	8	0.8	8.44e-03±3.45e-04
5	5	2	0.4	8.27e-03±8.30e-04
6	5	4	0.2	1.29e-02±1.39e-03
7	**5**	**6**	**0.8**	**7.36e-03±6.48e-04**
8	5	8	0.6	7.74e-03±6.88e-04
9	8	2	0.6	7.37e-03±8.89e-04
10	8	4	0.8	7.92e-03±8.30e-04
11	8	6	0.2	1.02e-02±2.27e-03
12	8	8	0.4	8.48e-03±1.38e-03
13	10	2	0.8	8.82e-03±2.32e-03
14	10	4	0.6	8.53e-03±1.81e-03
15	10	6	0.4	9.25e-03±2.14e-03
16	10	8	0.2	8.86e-03±8.64e-04

Table 4.14: $L_{16}(4^3)$ The Taguchi's experimental result of the EEG dataset.

No.	k	m	θ_{soma}	MSE(Average±Std)
1	2	4	0.2	2.91e-02±1.18e-04
2	2	6	0.4	1.92e-02±1.51e-04
3	2	8	0.6	1.62e-02±2.84e-04
4	2	10	0.8	1.54e-02±2.63e-04
5	5	4	0.4	1.55e-02±3.37e-04
6	5	6	0.2	1.77e-02±4.45e-04
7	5	8	0.8	1.52e-02±3.15e-04
8	**5**	**10**	**0.6**	**1.51e-02±3.32e-04**
9	8	4	0.6	1.60e-02±1.09e-03
10	8	6	0.8	1.58e-02±7.17e-04
11	8	8	0.2	1.64e-02±5.66e-04
12	8	10	0.4	1.60e-02±9.89e-04
13	10	4	0.8	1.74e-02±1.94e-03
14	10	6	0.6	1.67e-02±1.78e-03
15	10	8	0.4	1.63e-02±1.05e-03
16	10	10	0.2	1.68e-02±1.27e-03

Table 4.15: $L_{16}(4^3)$ The Taguchi's experimental result of the MackeyGlass dataset.

No.	k	m	θ_{soma}	MSE(Average±Std)
1	2	4	0.2	1.48e-02±1.05e-04
2	2	6	0.4	6.38e-03±1.56e-04
3	2	8	0.6	2.91e-03±2.01e-04
4	2	10	0.8	1.64e-03±1.85e-04
5	5	4	0.4	1.52e-03±3.54e-04
6	5	6	0.2	4.58e-03±2.64e-04
7	5	8	0.8	1.10e-03±4.03e-04
8	**5**	**10**	**0.6**	**9.67e-04±3.41e-04**
9	8	4	0.6	1.89e-03±9.97e-04
10	8	6	0.8	2.38e-03±1.66e-03
11	8	8	0.2	2.27e-03±5.34e-04
12	8	10	0.4	1.32e-03±4.93e-04
13	10	4	0.8	3.54e-03±2.26e-03
14	10	6	0.6	2.61e-03±2.27e-03
15	10	8	0.4	2.09e-03±1.46e-03
16	10	10	0.2	1.74e-03±5.37e-04

Table 4.16: $L_{16}(4^3)$ The Taguchi's experimental result of the Tourism dataset.

No.	k	m	θ_{soma}	MSE(Average±Std)
1	2	6	0.2	9.80e-03±8.17e-04
2	2	8	0.4	7.55e-03±5.03e-04
3	**2**	**10**	**0.6**	**7.38e-03±5.23e-04**
4	2	12	0.8	7.70e-03±9.54e-04
5	5	6	0.4	8.97e-03±2.77e-03
6	5	8	0.2	8.31e-03±1.11e-03
7	5	10	0.8	9.17e-03±2.45e-03
8	5	12	0.6	8.59e-03±1.63e-03
9	8	6	0.6	1.04e-02±3.52e-03
10	8	8	0.8	1.06e-02±4.50e-03
11	8	10	0.2	8.99e-03±1.67e-03
12	8	12	0.4	9.58e-03±2.36e-03
13	10	6	0.8	1.20e-02±5.94e-03
14	10	8	0.6	1.26e-02±5.85e-03
15	10	10	0.4	1.13e-02±4.99e-03
16	10	12	0.2	9.38e-03±2.35e-03

Table 4.17: $L_{16}(4^3)$ The Taguchi's experimental result of the Chaos-01 dataset.

No.	k	m	θ_{soma}	MSE(Average±Std)
1	2	4	0.2	1.31e-01±2.44e-03
2	2	6	0.4	1.18e-01±2.54e-03
3	2	8	0.6	1.05e-01±2.18e-03
4	2	10	0.8	9.20e-02±3.04e-03
5	5	4	0.4	7.03e-02±6.65e-03
6	5	6	0.2	1.04e-01±6.04e-03
7	**5**	**8**	**0.8**	**4.49e-02±9.21e-03**
8	5	10	0.6	5.08e-02±1.09e-02
9	8	4	0.6	4.93e-02±2.08e-02
10	8	6	0.8	5.00e-02±1.88e-02
11	8	8	0.2	8.20e-02±9.48e-03
12	8	10	0.4	5.68e-02±1.72e-02
13	10	4	0.8	6.74e-02±2.17e-02
14	10	6	0.6	5.17e-02±1.86e-02
15	10	8	0.4	5.37e-02±1.92e-02
16	10	10	0.2	7.63e-02±1.09e-02

Table 4.18: $L_{16}(4^3)$ The Taguchi's experimental result of the Chaos-02 dataset.

No.	k	m	θ_{soma}	MSE(Average±Std)
1	2	4	0.2	1.26e-01±1.96e-03
2	2	6	0.4	1.13e-01±2.03e-03
3	2	8	0.6	1.00e-01±1.94e-03
4	2	10	0.8	8.87e-02±1.83e-03
5	5	4	0.4	6.48e-02±5.15e-03
6	5	6	0.2	9.45e-02±3.88e-03
7	**5**	**8**	**0.8**	**3.55e-02±6.70e-03**
8	5	10	0.6	4.65e-02±6.68e-03
9	8	4	0.6	4.47e-02±1.60e-02
10	8	6	0.8	4.05e-02±1.27e-02
11	8	8	0.2	7.38e-02±6.06e-03
12	8	10	0.4	4.59e-02±9.82e-03
13	10	4	0.8	4.64e-02±1.42e-02
14	10	6	0.6	4.34e-02±1.55e-02
15	10	8	0.4	4.73e-02±1.28e-02
16	10	10	0.2	6.48e-02±6.13e-03

Table 4.19: $L_{16}(4^3)$ The Taguchi's experimental result of the Chaos-03 dataset.

No.	k	m	θ_{soma}	MSE(Average±Std)
1	2	4	0.2	8.92e-02±1.58e-03
2	2	6	0.4	8.20e-02±1.47e-03
3	2	8	0.6	7.11e-02±1.34e-03
4	2	10	0.8	6.14e-02±1.79e-03
5	5	4	0.4	4.44e-02±4.96e-03
6	5	6	0.2	6.20e-02±2.29e-03
7	**5**	**8**	**0.8**	**2.63e-02±7.04e-03**
8	5	10	0.6	3.49e-02±6.70e-03
9	8	4	0.6	3.42e-02±9.92e-03
10	8	6	0.8	3.22e-02±9.08e-03
11	8	8	0.2	4.93e-02±5.75e-03
12	8	10	0.4	3.46e-02±9.08e-03
13	10	4	0.8	3.81e-02±1.00e-02
14	10	6	0.6	3.49e-02±1.14e-02
15	10	8	0.4	3.64e-02±1.09e-02
16	10	10	0.2	4.49e-02±6.02e-03

Table 4.20: $L_{16}(4^3)$ The Taguchi's experimental result of the Chaos-04 dataset.

No.	k	m	θ_{soma}	MSE(Average±Std)
1	2	4	0.2	7.00e-02±1.41e-03
2	2	6	0.4	6.25e-02±1.56e-03
3	2	8	0.6	5.54e-02±1.74e-03
4	2	10	0.8	4.72e-02±1.70e-03
5	5	4	0.4	3.32e-02±5.15e-03
6	5	6	0.2	4.94e-02±2.82e-03
7	5	8	0.8	2.34e-02±8.21e-03
8	**5**	**10**	**0.6**	**2.18e-02±4.22e-03**
9	8	4	0.6	2.96e-02±9.10e-03
10	8	6	0.8	3.04e-02±7.55e-03
11	8	8	0.2	3.75e-02±4.45e-03
12	8	10	0.4	2.80e-02±8.66e-03
13	10	4	0.8	4.04e-02±8.96e-03
14	10	6	0.6	3.45e-02±9.49e-03
15	10	8	0.4	3.10e-02±1.10e-02
16	10	10	0.2	3.50e-02±7.69e-03

Table 4.21: Accuracy comparison of three models on eight classification datasets.

Datasets	Model	Max (%)	Min (%)	Average±Std (%)	p value
Breast	MLP	97.71	94.00	95.91±0.74	2.40e-03
	DT	96.57	91.14	94.21±1.17	1.16e-06
	line-SVM	96.86	93.43	95.91±0.88	3.33e-02
	rbf-SVM	97.14	95.14	96.05±0.59	5.48e-02
	poly-SVM	97.14	**95.43**	96.27±0.59	3.92e-01
	DNM	97.43	93.43	95.51±1.08	5.50e-03
	AISDNM	**98.00**	94.57	**96.37±0.83**	-
Glass	MLP	96.26	85.98	91.81±2.84	1.10e-03
	DT	96.26	82.24	90.84±3.16	3.17e-05
	line-SVM	96.26	86.92	91.37±2.08	1.89e-05
	rbf-SVM	96.26	85.98	91.03±2.62	1.88e-04
	poly-SVM	**97.20**	**88.79**	92.93±2.30	3.97e-02
	DNM	**97.20**	85.05	91.90±2.80	2.30e-03
	AISDNM	**97.20**	**88.79**	**93.96±1.97**	-
Haberman	MLP	78.43	**69.93**	73.73±2.17	5.30e-03
	DT	78.43	62.09	68.26±4.06	2.31e-06
	line-SVM	79.74	68.63	72.85±2.94	2.60e-03
	rbf-SVM	78.43	67.97	73.51±2.71	5.90e-03
	poly-SVM	77.12	69.28	73.09±2.29	5.01e-03
	DNM	80.39	**69.93**	73.88±2.45	2.30e-03
	AISDNM	**81.05**	**69.93**	**75.19±2.44**	-
Iris	MLP	**98.67**	82.67	92.76±3.43	8.00e-03
	DT	**98.67**	84.00	92.98±3.45	3.74e-03
	line-SVM	73.33	60.00	66.18±3.39	8.60e-07
	rbf-SVM	**98.67**	86.67	93.78±2.87	7.89e-02
	poly-SVM	**98.67**	81.33	94.58±3.21	4.95e-01
	DNM	97.33	86.67	93.69±2.24	3.36e-02
	AISDNM	97.33	**89.33**	**94.76±1.71**	-
Thyroid	MLP	96.30	79.63	89.72±3.79	2.01e-05
	DT	**97.22**	86.11	91.05±2.25	1.66e-05
	line-SVM	86.11	69.44	78.15±3.81	9.01e-07
	rbf-SVM	93.52	79.63	87.22±3.22	8.71e-07
	poly-SVM	94.44	82.41	89.51±2.85	2.96e-06
	DNM	**97.22**	87.04	92.69±2.37	5.80e-03
	AISDNM	**97.22**	**90.74**	**94.20±1.83**	-
Wine	MLP	98.88	89.89	94.72±2.44	1.78e-01
	DT	95.51	84.27	91.24±2.76	1.84e-05
	line-SVM	**100.00**	**91.01**	96.25±1.99	9.50e-01
	rbf-SVM	**100.00**	**91.01**	**97.38±1.83**	9.99e-01
	poly-SVM	98.88	**91.01**	95.81±2.38	8.30e-01
	DNM	97.75	85.39	91.54±3.35	7.84e-05
	AISDNM	98.88	89.89	95.17±2.21	-
Rice	MLP	93.23	84.46	90.38±2.06	2.52e-06
	DT	90.39	87.30	89.07±0.76	9.08e-07
	line-SVM	93.75	91.97	92.90±0.43	4.29e-01
	rbf-SVM	93.65	91.97	92.86±0.47	4.10e-01
	poly-SVM	93.96	**92.23**	92.83±0.40	1.68e-01
	DNM	93.70	89.87	92.00±0.95	9.67e-06
	AISDNM	**94.12**	91.86	**92.92±0.46**	-
Heart	MLP	79.33	60.00	70.09±5.05	1.00e-06
	DT	84.00	73.33	78.13±2.75	3.91e-05
	line-SVM	86.67	73.33	81.07±2.95	7.78e-02
	rbf-SVM	80.67	66.00	73.51±3.28	8.89e-07
	poly-SVM	80.67	68.67	74.49±3.12	1.63e-06
	DNM	86.67	72.00	81.09±4.08	2.03e-01
	AISDNM	**88.00**	**74.67**	**82.07±2.80**	-

Table 4.22: Additional comparison of three models on eight classification datasets.

Datasets	Model	Sensitivity (%) Average±Std	Specificity (%) Average±Std	$F_{measure}$ Average±Std	K Average±Std	AUC Average±Std
Breast	MLP	97.17±0.86	93.49±2.23	0.969±0.006	0.909±0.016	0.9917±0.003
	DT	95.08±1.53	92.55±2.92	0.956±0.009	0.872±0.026	0.9382±0.014
	line-SVM	97.86±0.71	92.31±2.36	0.969±0.007	0.909±0.019	0.9509±0.011
	rbf-SVM	97.45±0.67	93.36±1.19	0.970±0.005	0.912±0.013	0.9540±0.007
	poly-SVM	97.61±0.63	93.72±1.92	0.971±0.004	0.917±0.014	0.9567±0.009
	DNM	97.16±0.89	92.37±2.72	0.966±0.008	0.900±0.024	0.9929±0.003
	AISDNM	**97.31±0.81**	**94.58±2.04**	**0.972±0.006**	**0.919±0.019**	**0.9945±0.002**
Glass	MLP	76.64±8.36	96.65±2.47	0.816±0.061	0.764±0.079	**0.9691±0.023**
	DT	95.99±2.56	74.41±11.85	0.941±0.021	0.733±0.090	0.8520±0.056
	line-SVM	96.94±3.04	73.12±8.50	0.945±0.013	0.739±0.064	0.8503±0.037
	rbf-SVM	97.04±2.62	72.30±8.87	0.942±0.018	0.738±0.074	0.8467±0.042
	poly-SVM	96.27±2.04	82.22±7.20	**0.954±0.016**	0.800±0.063	0.8925±0.037
	DNM	95.35±2.59	81.80±9.63	0.831±0.053	0.778±0.071	0.9539±0.035
	AISDNM	**97.06±2.50**	**84.18±6.91**	0.869±0.042	**0.829±0.054**	0.9545±0.032
Haberman	MLP	97.77±2.23	8.54±7.21	0.845±0.013	0.083±0.067	0.6187±0.067
	DT	81.44±7.46	**32.72±8.46**	0.788±0.034	0.150±0.075	0.5708±0.034
	line-SVM	**99.68±0.70**	0.00±0.00	0.843±0.019	0.005±0.010	0.4984±0.004
	rbf-SVM	98.61±2.11	2.99±5.43	0.846±0.018	0.021±0.047	0.5080±0.018
	poly-SVM	97.89±2.70	4.89±6.70	0.842±0.016	0.036±0.056	0.5139±0.022
	DNM	93.12±3.51	23.40±9.96	0.838±0.016	0.197±0.074	**0.7008±0.034**
	AISDNM	93.40±3.28	24.20±8.86	**0.847±0.017**	**0.212±0.078**	0.6886±0.036
Iris	MLP	90.94±7.85	93.78±4.64	0.891±0.053	0.837±0.076	0.9832±0.014
	DT	95.23±3.84	88.30±7.11	0.948±0.026	0.839±0.078	0.9177±0.040
	line-SVM	**97.83±5.90**	4.14±11.35	0.793±0.024	0.021±0.063	0.5099±0.030
	rbf-SVM	92.35±3.40	97.01±6.13	0.952±0.023	0.863±0.061	0.9468±0.032
	poly-SVM	92.88±4.92	**98.41±2.81**	**0.958±0.027**	0.880±0.065	0.9564±0.024
	DNM	91.65±6.19	94.67±4.43	0.906±0.031	0.858±0.047	0.9858±0.010
	AISDNM	94.09±4.90	95.12±2.56	0.919±0.030	**0.880±0.041**	**0.9910±0.009**
Thyroid	MLP	99.87±0.39	66.27±10.42	0.932±0.025	0.728±0.096	0.9625±0.028
	DT	94.28±3.33	83.38±7.83	0.937±0.016	0.781±0.058	0.8883±0.034
	line-SVM	**100.00±0.00**	28.16±8.70	0.864±0.025	0.350±0.097	0.6408±0.044
	rbf-SVM	**100.00±0.00**	57.35±8.10	0.916±0.022	0.651±0.079	0.7867±0.040
	poly-SVM	**100.00±0.00**	64.44±7.92	0.931±0.020	0.717±0.073	0.8222±0.040
	DNM	97.45±2.48	81.43±8.72	0.949±0.017	0.818±0.060	0.9639±0.040
	AISDNM	97.94±2.13	**85.29±5.43**	**0.960±0.013**	**0.856±0.045**	**0.9710±0.020**
Wine	MLP	90.85±4.56	97.48±2.59	0.934±0.030	0.889±0.050	0.9888±0.010
	DT	87.27±6.02	93.95±3.99	0.890±0.033	0.817±0.055	0.9061±0.029
	line-SVM	91.01±4.85	**99.82±0.54**	0.951±0.028	0.921±0.043	0.9542±0.025
	rbf-SVM	**95.03±4.40**	99.08±1.18	**0.967±0.022**	**0.945±0.037**	0.9705±0.021
	poly-SVM	93.05±4.79	97.61±2.93	0.947±0.029	0.912±0.049	0.9533±0.026
	DNM	86.28±5.87	95.17±5.20	0.892±0.043	0.823±0.070	0.9711±0.017
	AISDNM	91.21±4.05	98.01±2.39	0.939±0.028	0.899±0.046	**0.9901±0.007**
Rice	MLP	92.75±2.69	87.18±4.70	91.727±1.732	0.825±0.005	0.9644±0.014
	DT	90.32±1.17	87.42±1.46	90.401±0.666	**0.827±0.002**	0.8887±0.008
	line-SVM	93.91±0.91	91.56±0.97	93.796±0.392	0.819±0.001	0.9274±0.004
	rbf-SVM	93.97±0.62	91.37±1.01	93.773±0.415	0.819±0.001	0.9267±0.005
	poly-SVM	**94.29±0.86**	90.87±1.03	93.785±0.367	0.820±0.001	0.9258±0.004
	DNM	93.83±0.81	91.20±0.98	93.623±0.332	0.819±0.001	0.9778±0.002
	AISDNM	93.56±0.76	**92.08±1.26**	**93.799±0.402**	0.819±0.001	**0.9791±0.002**
Heart	MLP	82.63±5.40	41.21±11.38	78.481±2.591	0.277±0.039	0.6845±0.055
	DT	84.71±5.08	**64.41±9.27**	83.949±2.386	0.378±0.028	0.7456±0.037
	line-SVM	89.69±3.42	62.80±10.27	86.655±2.130	0.399±0.030	0.7625±0.043
	rbf-SVM	92.59±3.26	34.66±9.66	82.491±2.070	0.307±0.033	0.6362±0.039
	poly-SVM	83.26±4.53	56.11±7.86	81.595±2.530	0.338±0.029	0.6968±0.035
	DNM	**95.57±1.63**	50.11±11.87	87.377±2.579	0.387±0.045	0.8254±0.052
	AISDNM	91.94±3.24	61.54±7.55	**87.434±2.082**	**0.408±0.026**	**0.8398±0.030**

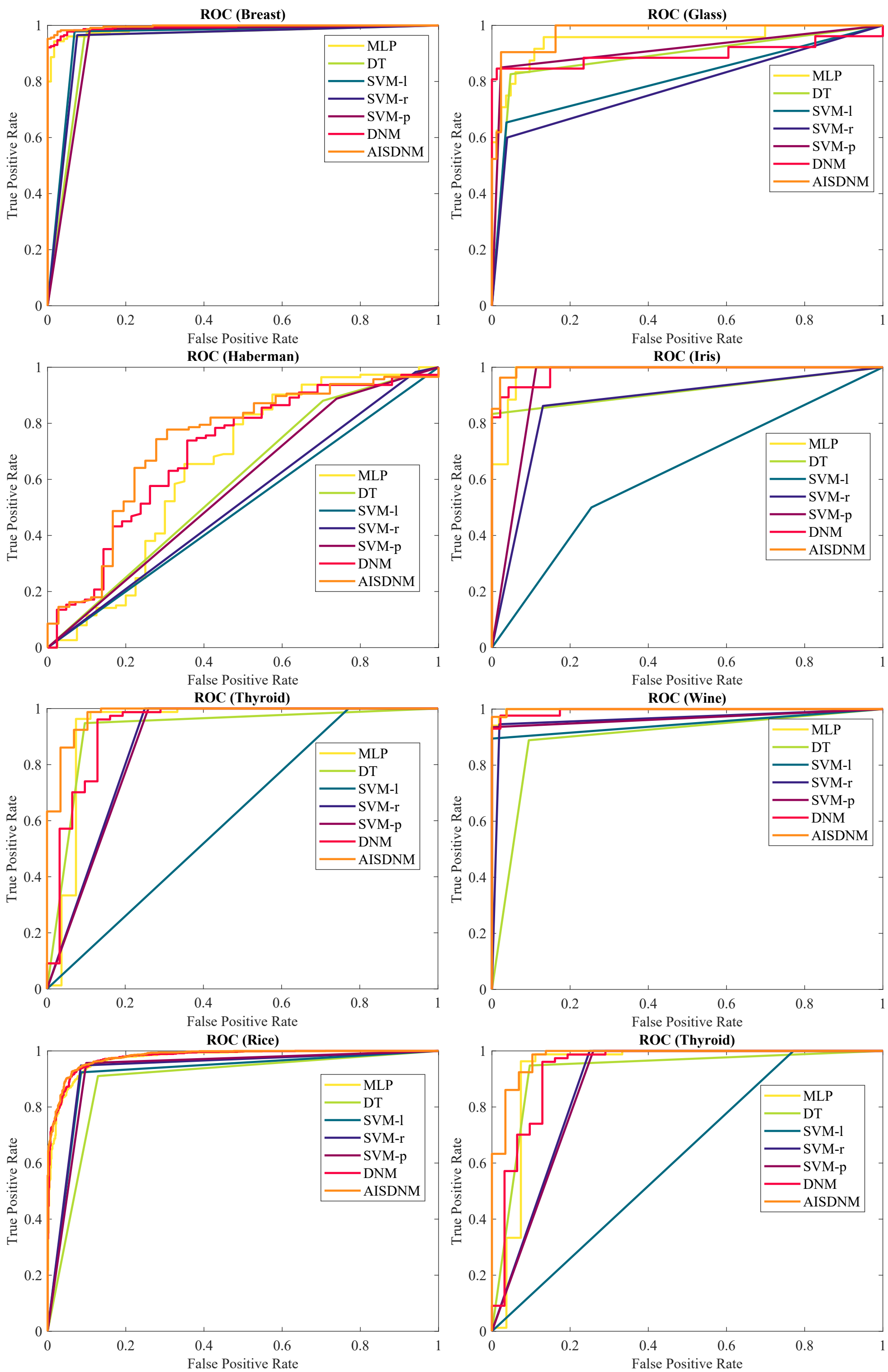

Figure 4.1: The ROCs of three models for eight classification datasets.

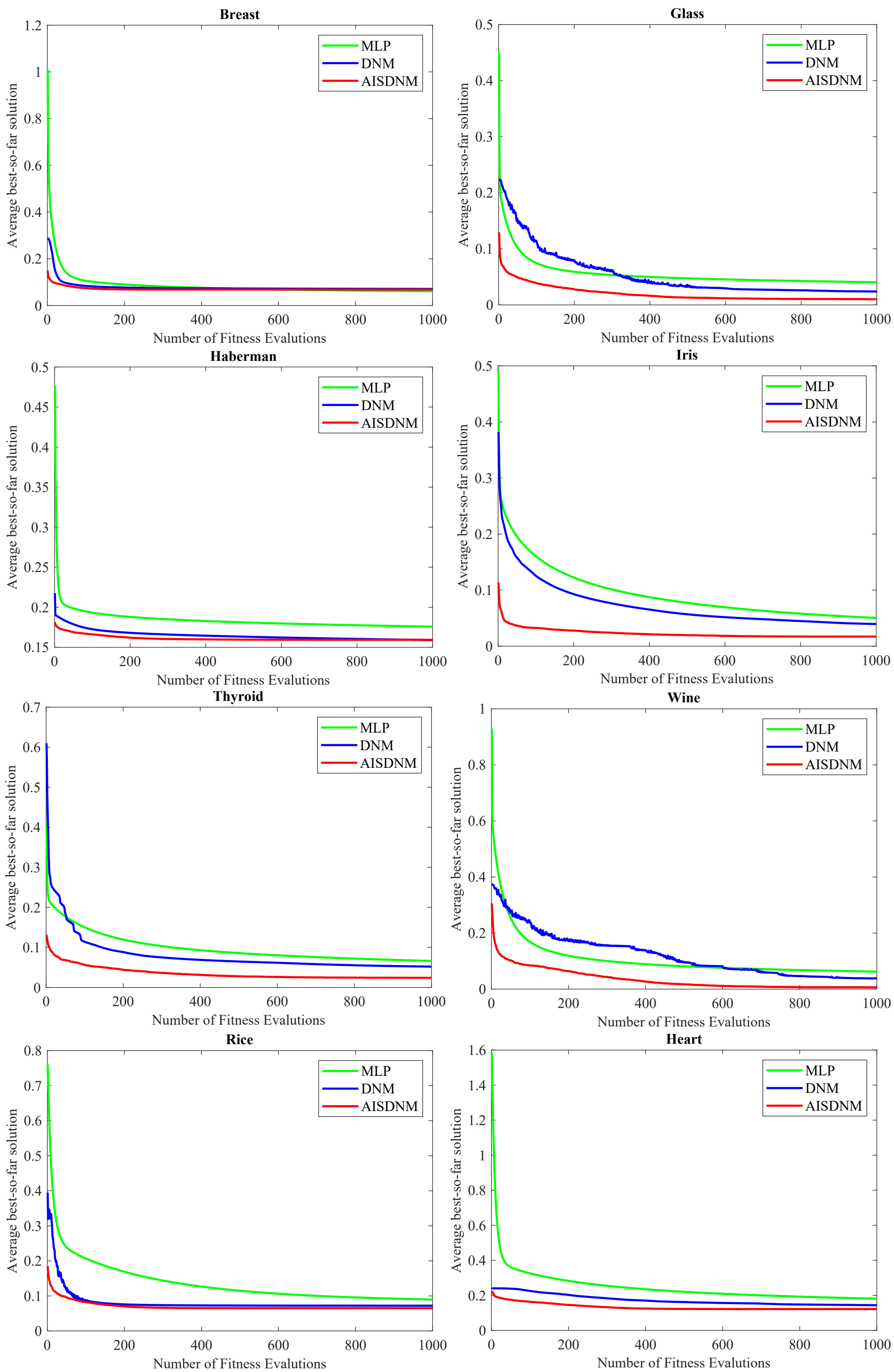

Figure 4.2: The convergence speeds of three models for eight classification datasets.

Table 4.23: Experimental results of the cross-validation methods on eight classification datasets.

Datasets	Model	5-fold CV (Acc)		10-fold CV (Acc)	
		Average±Std	p value	Average±Std	p value
Breast	MLP	96.0371±0.566	8.10e-03	96.0572±0.344	1.93e-04
	DT	94.2820±0.600	9.09e-07	94.0491±0.627	9.10e-07
	line-SVM	96.0644±0.104	7.58e-03	96.0920±0.100	2.46e-06
	rbf-SVM	**96.3258±0.141**	9.89e-01	96.4075±0.134	9.37e-01
	poly-SVM	96.3263±0.182	9.65e-01	**96.4933±0.141**	9.98e-01
	DNM	95.5612±0.314	1.00e-06	95.7965±0.363	1.93e-06
	AISDNM	96.1952±0.266	-	96.3655±0.185	-
Glass	MLP	92.0393±0.731	9.06e-07	91.8394±1.198	1.49e-06
	DT	92.3892±1.129	1.01e-06	92.2032±0.999	9.99e-07
	line-SVM	92.0035±0.704	1.01e-06	92.0622±0.724	1.00e-06
	rbf-SVM	92.5597±0.776	1.23e-06	92.5892±0.614	1.21e-06
	poly-SVM	93.3596±1.060	5.60e-05	93.3727±0.810	1.83e-05
	DNM	91.8489±2.930	7.06e-05	90.5835±2.688	1.50e-06
	AISDNM	**94.5901±0.945**	-	**94.7251±0.935**	-
Haberman	MLP	73.9554±0.544	1.00e-06	73.9870±0.748	2.45e-06
	DT	66.8870±2.100	9.12e-07	68.1852±1.767	9.10e-07
	line-SVM	73.3896±0.166	9.10e-07	73.3593±0.302	9.06e-07
	rbf-SVM	72.8447±0.469	9.10e-07	72.9426±0.427	9.09e-07
	poly-SVM	72.8332±0.611	9.12e-07	72.5852±0.577	9.08e-07
	DNM	74.1303±0.601	5.33e-06	75.2611±0.484	8.50e-02
	AISDNM	**75.1895±0.779**	-	**75.4241±0.416**	-
Iris	MLP	93.0000±2.128	2.10e-02	93.6222±1.676	8.00e-03
	DT	94.5778±1.144	9.91e-01	94.4667±1.038	4.20e-01
	line-SVM	66.8000±0.829	7.89e-07	66.4222±0.934	8.52e-07
	rbf-SVM	94.2667±0.851	9.84e-01	94.2222±0.708	3.42e-02
	poly-SVM	**96.2222±0.750**	1.00	**96.5333±0.664**	1.00
	DNM	77.4667±5.941	8.98e-07	75.1333±5.331	8.99e-07
	AISDNM	93.8889±0.911	-	94.4889±0.741	-
Thyroid	MLP	89.3488±1.663	8.85e-07	88.8205±1.555	9.06e-07
	DT	91.4574±1.390	1.30e-06	92.5647±1.187	2.20e-06
	line-SVM	80.0000±0.743	8.56e-07	80.0482±0.533	9.07e-07
	rbf-SVM	88.5736±0.435	8.20e-07	89.0226±0.312	9.00e-07
	poly-SVM	89.2093±0.628	8.08e-07	89.5379±0.554	8.97e-07
	DNM	91.0078±3.317	4.33e-06	90.5281±2.661	9.10e-07
	AISDNM	**94.5426±0.668**	-	**94.5611±0.626**	-
Wine	MLP	96.7469±1.212	1.00	96.7592±1.223	1.00
	DT	90.7774±1.683	9.10e-07	90.4267±1.441	9.08e-07
	line-SVM	97.3023±0.811	1.00	97.7153±0.517	1.00
	rbf-SVM	**98.3830±0.647**	1.00	**98.8996±0.419**	1.00
	poly-SVM	96.3749±1.092	1.00	96.3255±0.883	1.00
	DNM	79.4506±8.427	9.13e-07	81.6227±5.232	9.12e-07
	AISDNM	95.0757±1.497	-	95.2314±0.943	-
Rice	MLP	89.7918±0.838	8.87e-07	89.3526±0.369	5.44e-07
	DT	89.2896±0.348	8.96e-07	89.0647±0.127	5.79e-07
	line-SVM	92.3648±0.063	8.95e-07	92.3928±0.056	8.78e-07
	rbf-SVM	92.5057±0.068	1.04e-06	92.5144±0.040	8.81e-07
	poly-SVM	92.6185±0.094	5.53e-05	92.6544±0.056	7.20e-05
	DNM	92.5284±0.130	1.10e-06	92.3482±0.177	1.13e-05
	AISDNM	**92.7437±0.104**	-	**92.7445±0.072**	-
Heart	MLP	70.0705±2.648	9.13e-07	69.7949±2.382	9.12e-07
	DT	78.6973±1.970	9.12e-07	79.2871±1.700	9.09e-07
	line-SVM	82.1247±0.867	4.14e-05	82.5499±0.912	2.42e-05
	rbf-SVM	76.4470±0.930	9.10e-07	77.0227±0.840	9.09e-07
	poly-SVM	75.4451±1.358	9.12e-07	75.6370±1.185	9.12e-07
	DNM	83.4739±1.176	9.69e-01	83.4498±0.878	3.63e-02
	AISDNM	**83.0164±1.075**	-	**83.8714±1.048**	-

Table 4.24: Comparison of the AISDNM and the LCC on eight classification datasets.

Dataset	Acc of the AISDNM (%)	Acc of the LCC (%)
Breast	97.14	97.71
Glass	97.20	97.20
Haberman	73.86	76.47
Iris	97.33	98.67
Thyroid	97.22	96.30
Wine	94.38	97.75
Rice	93.33	93.28
Heart	85.33	86.67

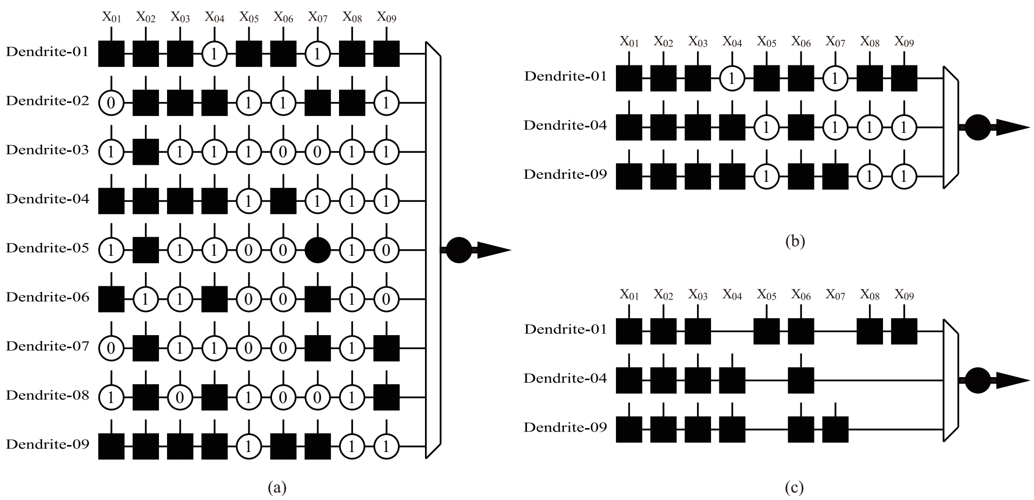

Figure 4.3: The evolution of the AISDNM structure on the Breast dataset.

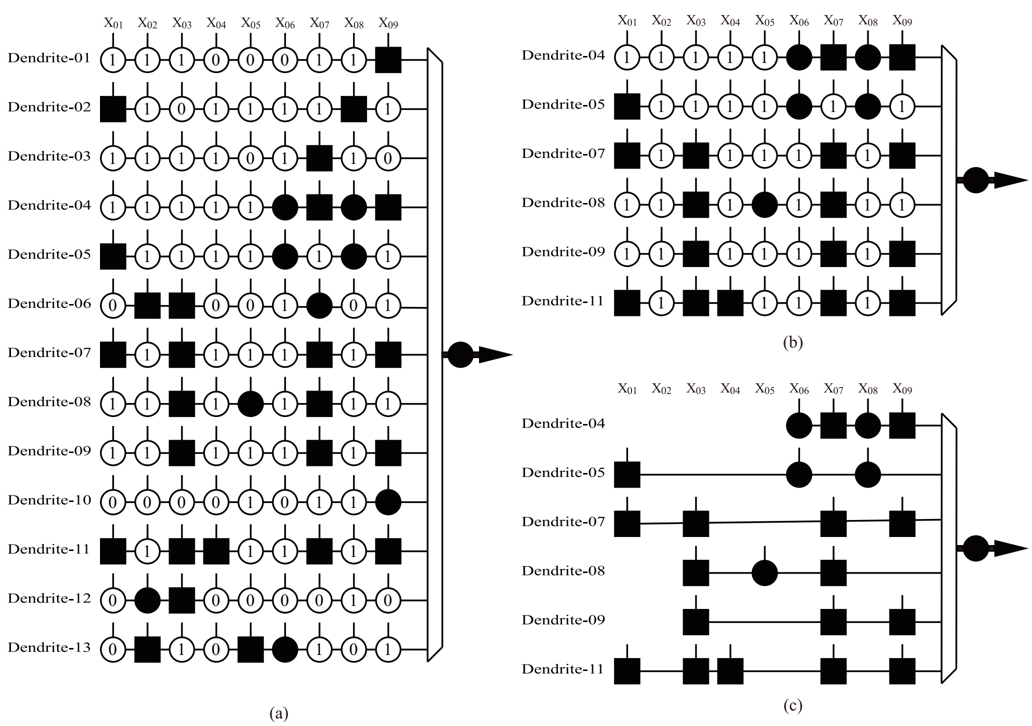

Figure 4.4: The evolution of the structure of the AISDNM on the Glass dataset.

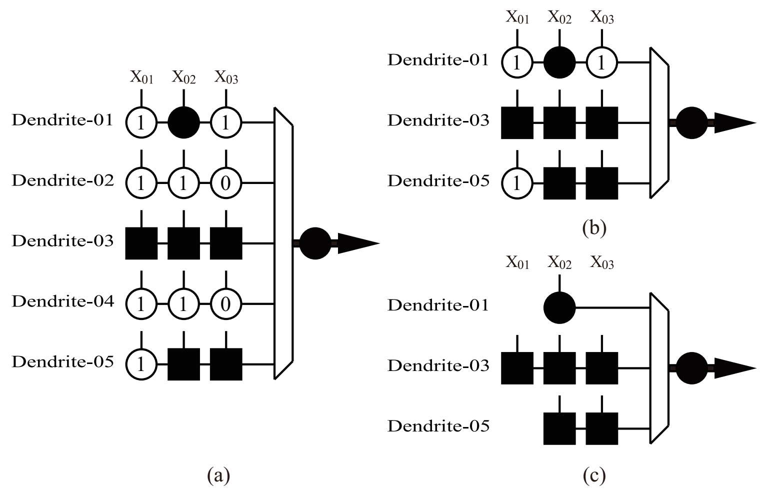

Figure 4.5: The evolution of the structure of the AISDNM on the Haberman dataset.

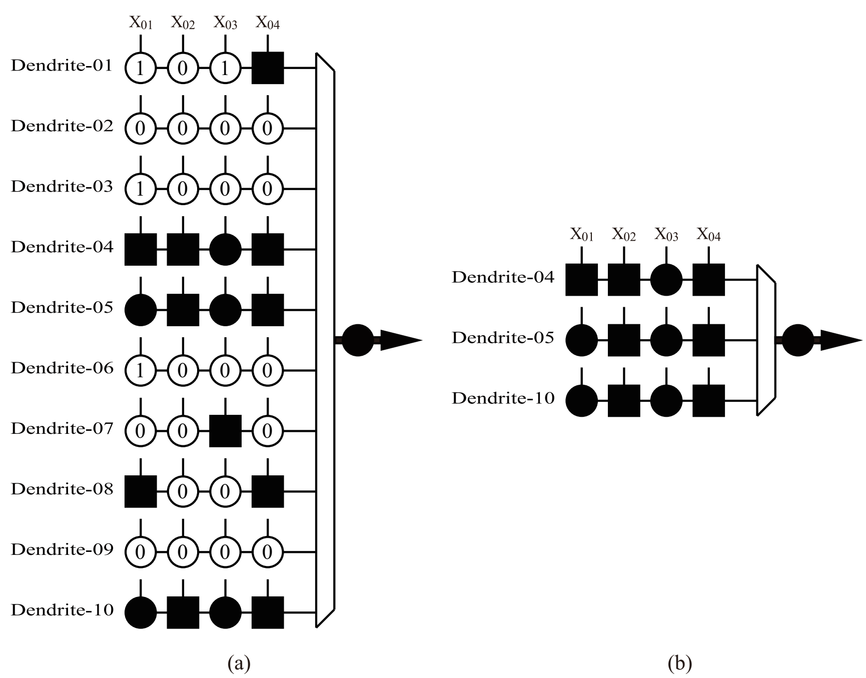

Figure 4.6: The evolution of the structure of the AISDNM on the Iris dataset.

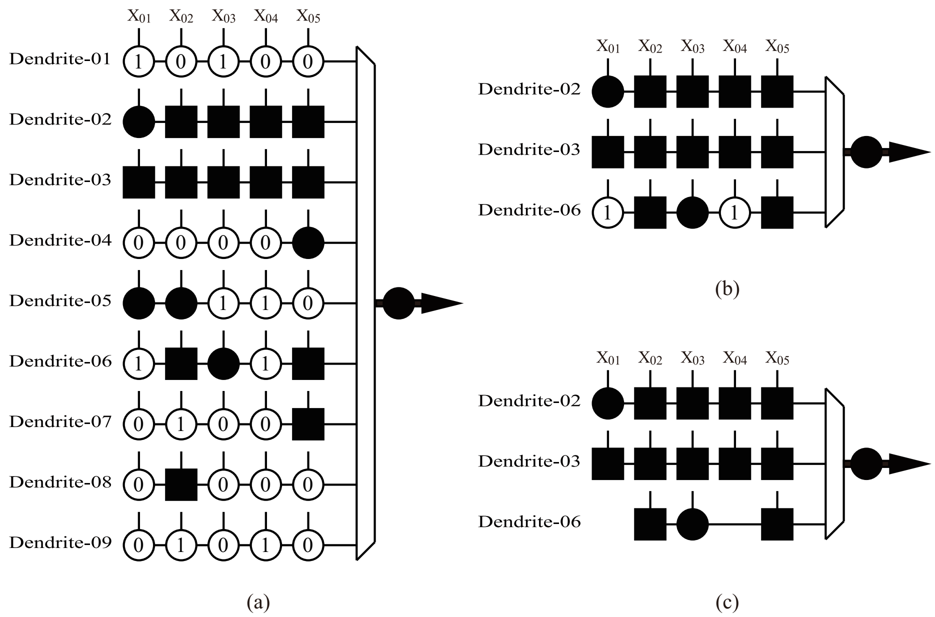

Figure 4.7: The evolution of the structure of the AISDNM on the Thyroid dataset.

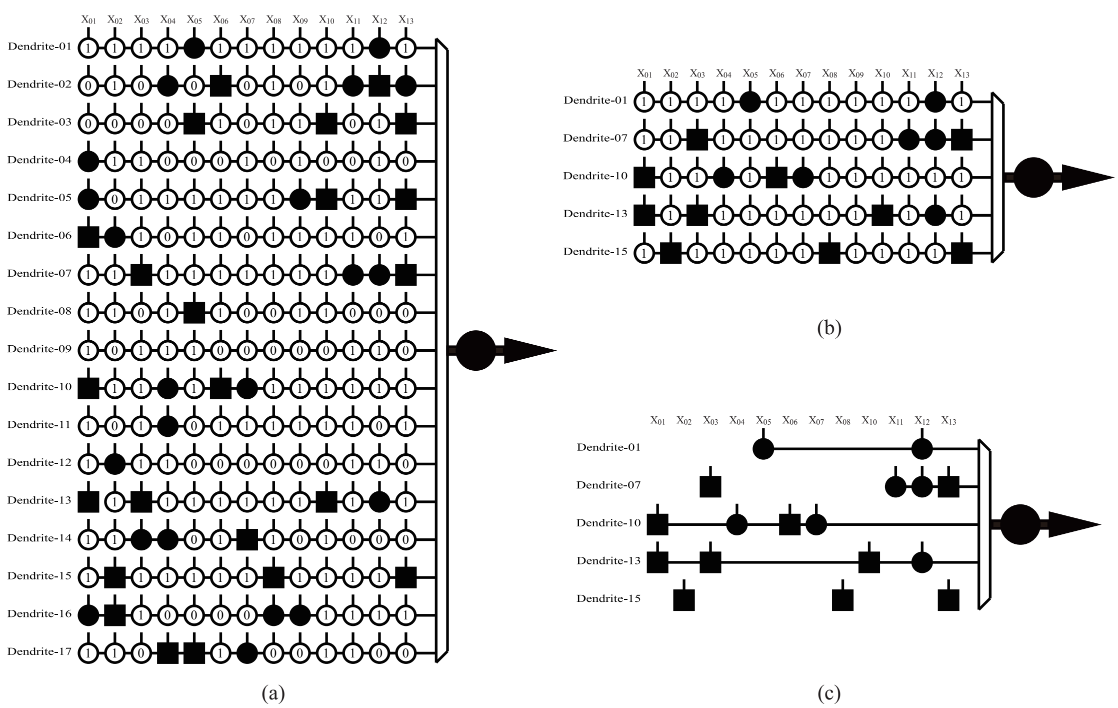

Figure 4.8: The evolution of the structure of the AISDNM on the Wine dataset.

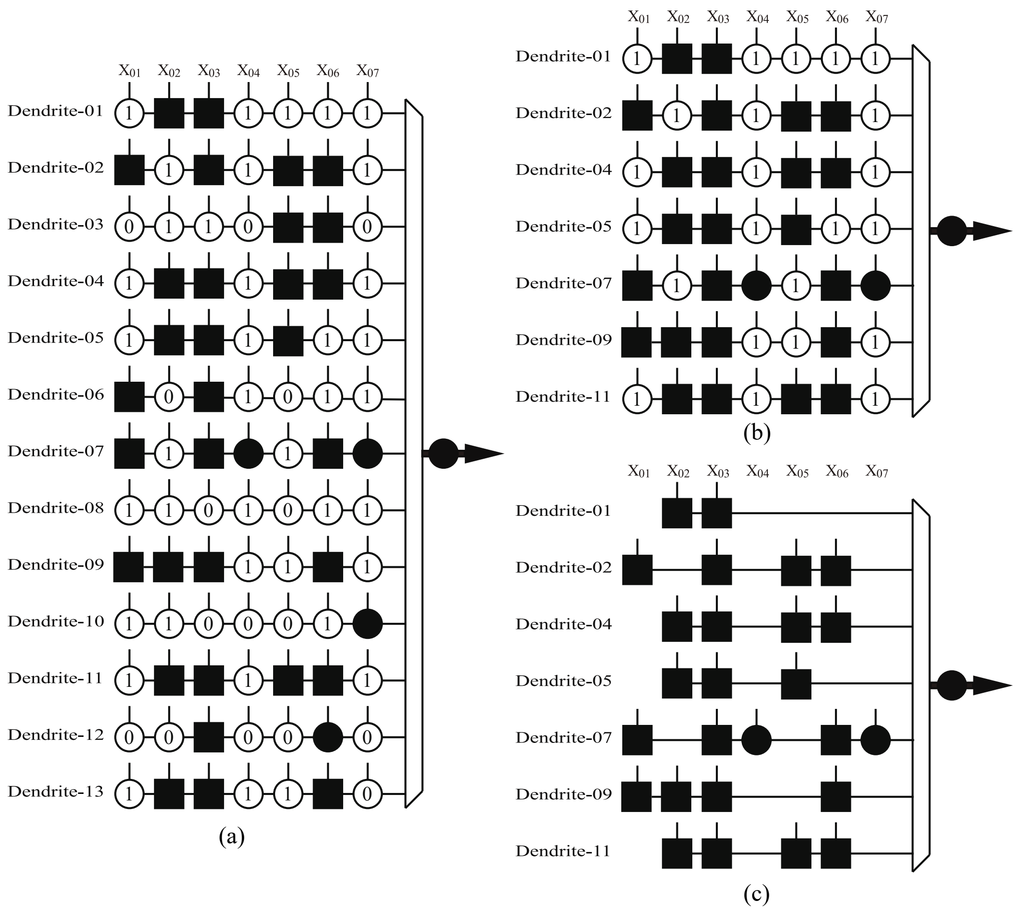

Figure 4.9: The evolution of the structure of the AISDNM on the Rice dataset.

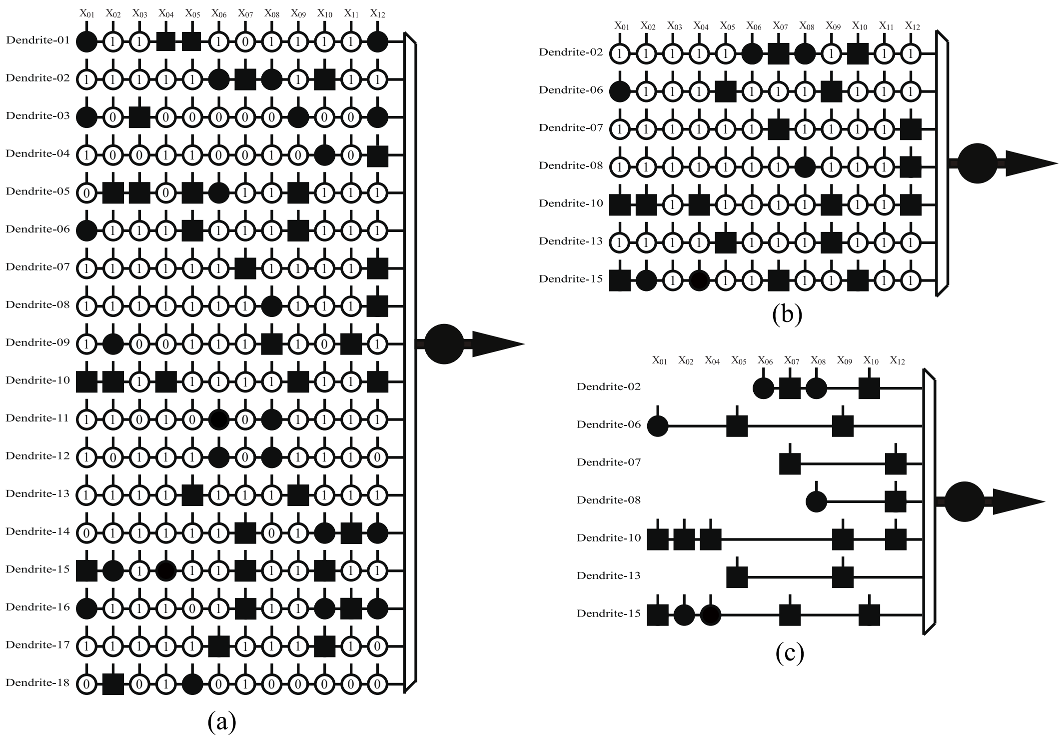

Figure 4.10: The evolution of the structure of the AISDNM on the Heart dataset.

Table 4.25: Comparison of DNM performance for prediction problems.

BoxJenkins								
Model	MSE	p	MAPE	p	MAE	p	R	p
MLP	2.40e-02±7.63e-03	9.13e-07	5.57e-02±3.67e-02	4.89e-06	1.22e-01±1.95e-02	9.13e-07	5.78e-01±2.20e-01	9.13e-07
DT	4.87e-03±8.82e-19	1.01e-06	**3.06e-03±1.76e-18**	1.00	5.51e-02±2.82e-17	9.13e-07	9.08e-01±6.78e-16	1.01e-06
line-SVM	7.58e-03±2.65e-18	9.13e-07	7.26e-02±4.23e-17	9.13e-07	6.99e-02±5.65e-17	9.13e-07	8.47e-01±2.26e-16	9.13e-07
rbf-SVM	4.27e-03±1.76e-18	1.03e-05	4.23e-02±7.06e-18	9.13e-07	5.08e-02±0.00e+00	5.51e-05	9.22e-01±3.39e-16	3.03e-05
poly-SVM	8.88e-03±1.76e-18	9.13e-07	2.57e-02±3.53e-18	9.13e-07	7.24e-02±4.23e-17	9.13e-07	8.71e-01±2.26e-16	9.13e-07
DNM	3.85e-02±8.80e-02	1.87e-04	2.17e-01±2.80e-01	1.63e-05	1.26e-01±1.35e-01	1.36e-04	8.77e-01±1.46e-01	9.52e-03
AISDNM	**3.74e-03±3.89e-04**	-	1.09e-02±6.00e-03	-	**4.89e-02±2.00e-03**	-	**9.30e-01±6.56e-03**	-
EEG								
Model	MSE	p	MAPE	p	MAE	p	R	p
MLP	1.37e-02±3.25e-03	9.13e-07	4.56e-02±2.18e-02	3.20e-04	9.27e-02±1.10e-02	9.13e-07	8.03e-01±5.24e-02	9.13e-07
DT	1.17e-02±3.53e-18	9.13e-07	2.94e-02±2.12e-17	1.67e-01	8.57e-02±2.82e-17	9.13e-07	8.30e-01±1.13e-16	9.13e-07
line-SVM	8.41e-03±3.53e-18	9.13e-07	6.99e-02±1.41e-17	9.13e-07	7.31e-02±4.23e-17	9.13e-07	**8.94e-01±5.65e-16**	1.00
rbf-SVM	7.76e-03±5.29e-18	6.40e-02	4.00e-02±7.06e-18	9.13e-07	7.04e-02±2.82e-17	8.25e-04	8.91e-01±6.78e-16	3.40e-01
poly-SVM	1.27e-02±8.82e-18	9.13e-07	**1.33e-02±7.06e-18**	1.00	8.83e-02±7.06e-17	9.13e-07	8.17e-01±1.13e-16	9.13e-07
DNM	2.03e-01±5.21e-02	9.13e-07	8.27e-01±2.36e-01	9.13e-07	4.07e-01±6.79e-02	9.13e-07	5.84e-01±1.88e-01	9.13e-07
AISDNM	**7.71e-03±2.04e-04**	-	2.85e-02±4.85e-03	-	**6.99e-02±7.35e-04**	-	8.91e-01±3.11e-03	-
MackeyGlass								
Model	MSE	p	MAPE	p	MAE	p	R	p
MLP	7.05e-03±5.73e-03	9.13e-07	2.83e-02±1.78e-02	9.13e-07	6.46e-02±2.52e-02	9.13e-07	9.37e-01±5.11e-02	9.13e-07
DT	1.61e-03±6.62e-19	9.13e-07	**2.76e-03±1.32e-18**	9.66e-01	3.17e-02±7.06e-18	9.13e-07	9.85e-01±5.65e-16	9.13e-07
line-SVM	2.08e-02±0.00e+00	9.13e-07	1.32e-01±5.65e-17	9.13e-07	1.29e-01±5.65e-17	9.13e-07	9.14e-01±5.65e-16	9.13e-07
rbf-SVM	1.55e-02±8.82e-18	9.13e-07	1.14e-01±4.23e-17	9.13e-07	1.12e-01±5.65e-17	9.13e-07	9.14e-01±0.00e+00	9.13e-07
poly-SVM	4.00e-03±2.65e-18	9.13e-07	4.40e-02±7.06e-18	9.13e-07	5.21e-02±2.12e-17	9.13e-07	9.69e-01±1.13e-16	9.13e-07
DNM	3.97e-02±3.59e-02	9.13e-07	1.92e-01±1.39e-01	9.13e-07	1.60e-01±7.47e-02	9.13e-07	9.33e-01±5.15e-02	9.13e-07
AISDNM	**4.51e-04±1.06e-04**	-	3.86e-03±3.02e-03	-	**1.65e-02±2.22e-03**	-	**9.96e-01±9.03e-04**	-
Tourism								
Model	MSE	p	MAPE	p	MAE	p	R	p
MLP	6.24e-03±1.91e-03	9.13e-07	5.04e-02±2.75e-02	3.30e-05	6.43e-02±1.00e-02	2.25e-06	6.19e-01±1.46e-01	2.04e-06
DT	5.95e-03±2.65e-18	9.13e-07	**1.74e-02±1.06e-17**	7.51e-01	6.39e-02±2.82e-17	9.13e-07	5.93e-01±1.13e-16	9.13e-07
line-SVM	8.21e-03±5.29e-18	9.13e-07	9.05e-02±0.00e+00	9.13e-07	7.20e-02±4.23e-17	9.13e-07	5.98e-01±3.39e-16	9.13e-07
rbf-SVM	4.91e-03±1.76e-18	9.13e-07	1.82e-02±7.06e-18	6.21e-01	5.49e-02±1.41e-17	1.51e-06	6.74e-01±1.13e-16	9.13e-07
poly-SVM	8.83e-03±1.76e-18	9.13e-07	8.60e-02±4.23e-17	9.13e-07	7.70e-02±4.23e-17	9.13e-07	7.40e-01±1.13e-16	1.24e-06
DNM	8.46e-03±1.63e-03	9.13e-07	7.50e-02±1.24e-02	9.13e-07	7.54e-02±7.30e-03	9.13e-07	6.75e-01±1.58e-01	2.96e-04
AISDNM	**3.68e-03±3.35e-04**	-	1.96e-02±1.31e-02	-	**5.02e-02±2.79e-03**	-	**7.61e-01±1.51e-02**	-

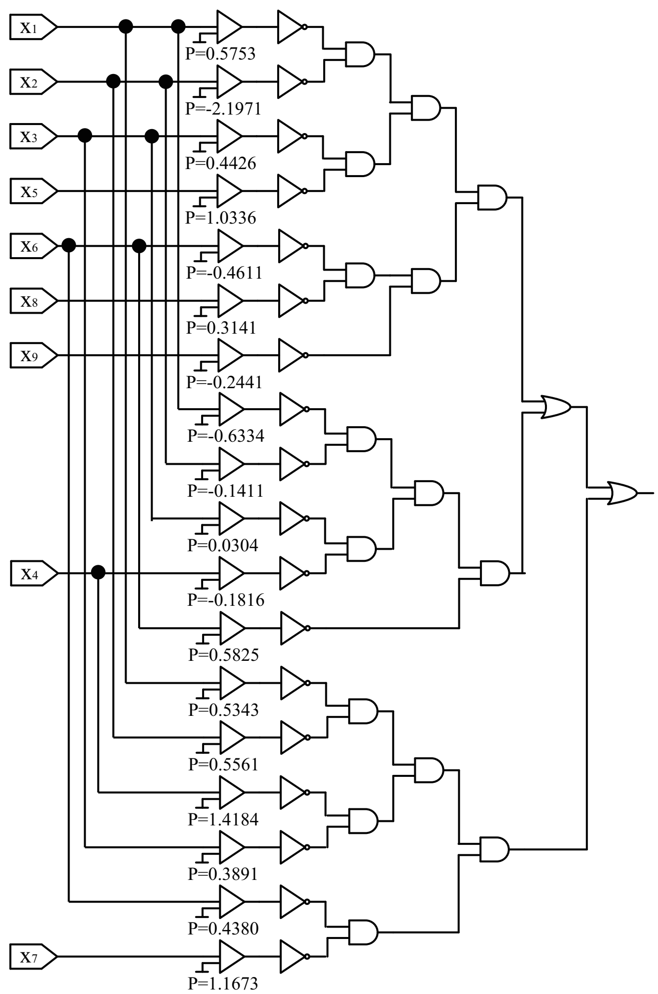

Figure 4.11: The logic circuits of the AISDNM on the Breast.

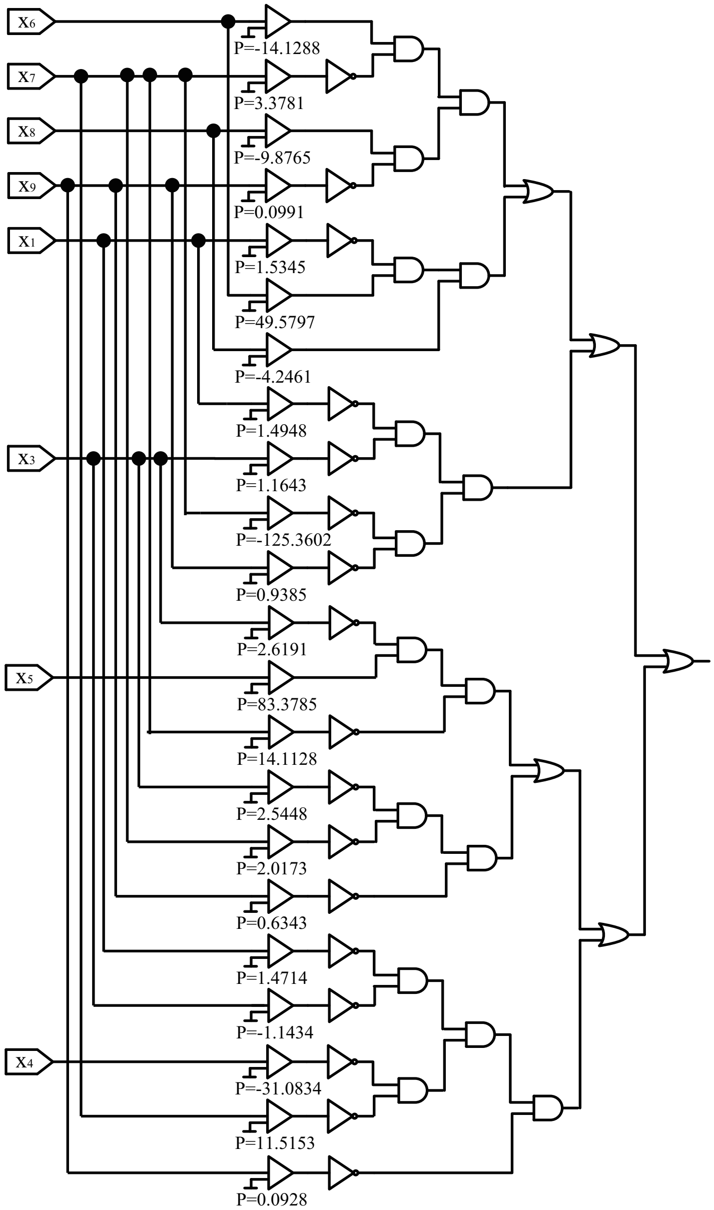

Figure 4.12: The logic circuits of the AISDNM on the Glass.

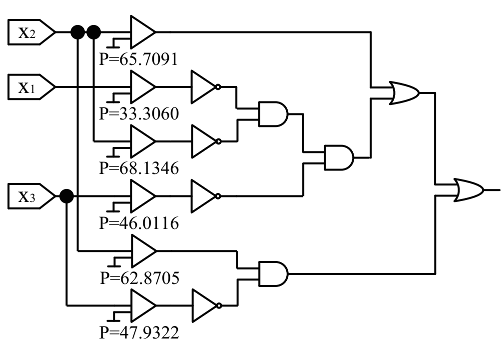

Figure 4.13: The logic circuits of the AISDNM on the Haberman.

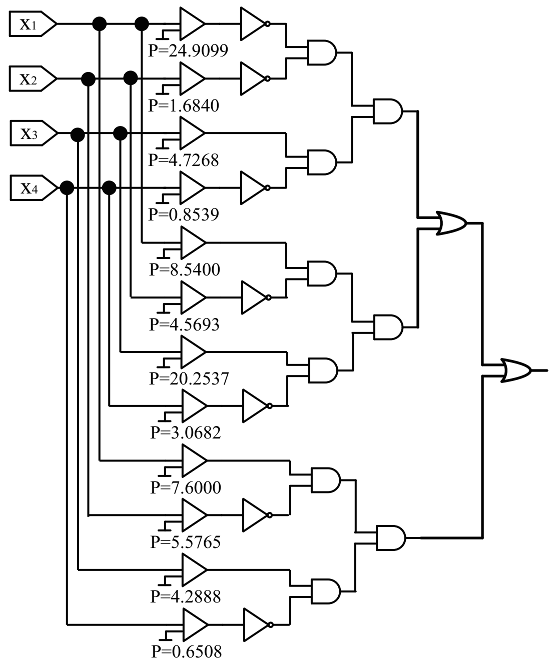

Figure 4.14: The logic circuits of the AISDNM on the Iris.

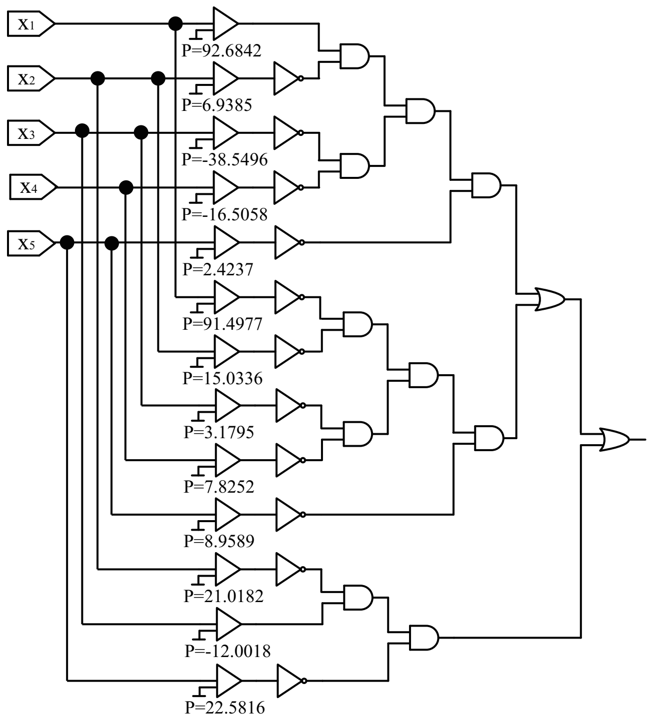

Figure 4.15: The logic circuits of the AISDNM on the Thyroid.

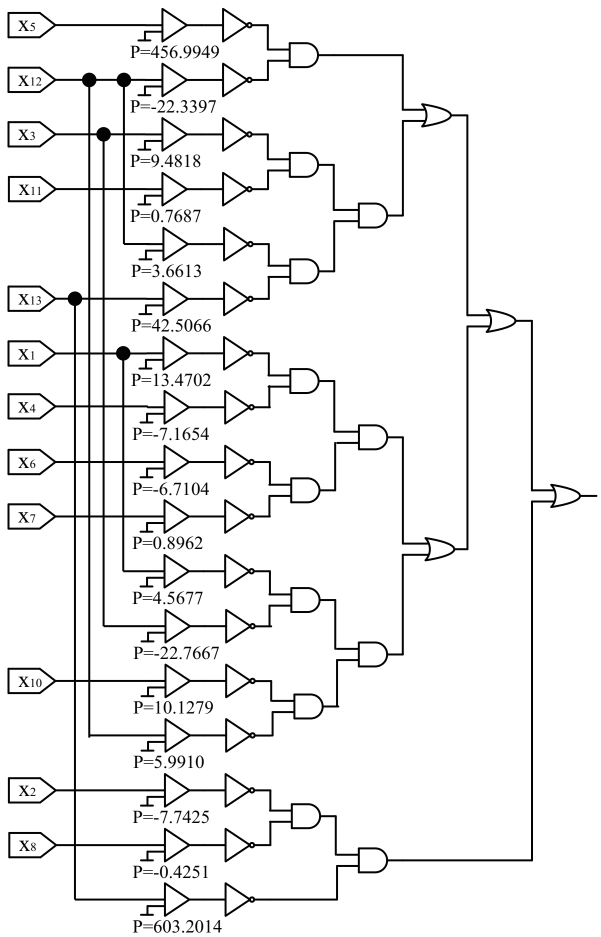

Figure 4.16: The logic circuits of the AISDNM on the Wine.

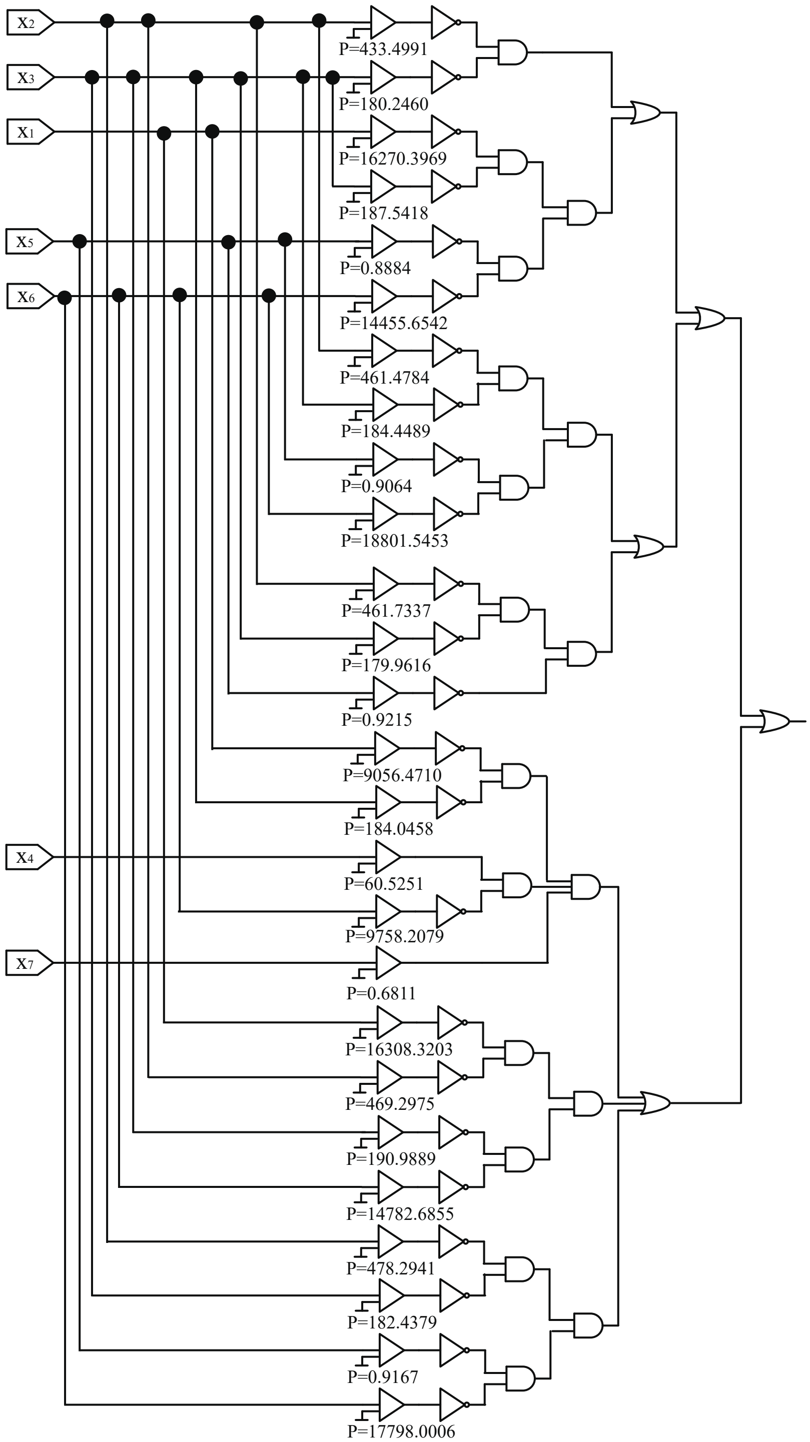

Figure 4.17: The logic circuits of the AISDNM on the Rice.

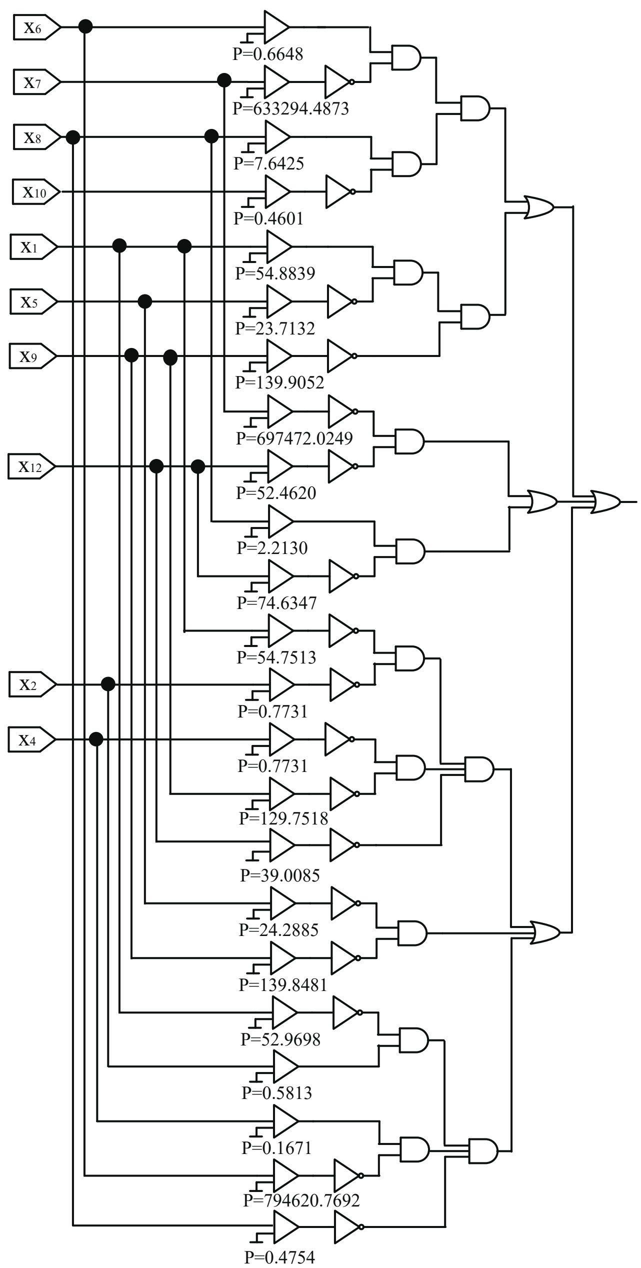

Figure 4.18: The logic circuits of the AISDNM on the Heart.

Table 4.26: Comparison of DNM performance for prediction problems.

Model	MSE	p	MAPE	p	MAE	p	R	p
Chaos-01								
MLP	8.04e-02±2.77e-02	2.25e-06	2.18e-01±5.64e-02	1.12e-06	2.40e-01±5.12e-02	2.04e-06	5.90e-01±2.07e-01	2.25e-06
DT	8.46e-02±1.41e-17	9.13e-07	1.92e-01±5.65e-17	9.13e-07	2.38e-01±1.13e-16	9.13e-07	5.74e-01±4.52e-16	9.13e-07
line-SVM	1.31e-01±0.00e+00	9.13e-07	3.16e-01±2.82e-16	9.13e-07	3.21e-01±0.00e+00	9.13e-07	5.33e-02±1.41e-17	9.13e-07
rbf-SVM	4.80e-02±1.41e-17	2.54e-05	2.04e-01±0.00e+00	9.13e-07	1.97e-01±5.65e-17	9.13e-07	**9.96e-01±7.90e-16**	1.00
poly-SVM	9.18e-02±4.23e-17	9.13e-07	2.42e-01±1.69e-16	9.13e-07	2.60e-01±0.00e+00	9.13e-07	5.28e-01±3.39e-16	9.13e-07
DNM	3.82e-02±8.66e-03	9.06e-02	1.69e-01±8.75e-02	1.63e-05	1.66e-01±1.57e-02	1.47e-04	9.60e-01±1.85e-02	1.00
AISDNM	**3.48e-02±1.14e-02**	-	**7.22e-02±3.27e-02**	-	**1.44e-01±2.03e-02**	-	8.89e-01±5.82e-02	-
Chaos-02								
Model	MSE	p	MAPE	p	MAE	p	R	p
MLP	8.83e-02±2.83e-02	1.37e-06	2.75e-01±7.08e-02	1.37e-06	2.51e-01±5.13e-02	2.04e-06	5.48e-01±2.03e-01	1.01e-06
DT	7.96e-02±2.82e-17	9.13e-07	2.63e-01±1.69e-16	9.13e-07	2.30e-01±1.41e-16	9.13e-07	6.26e-01±3.39e-16	9.13e-07
line-SVM	1.33e-01±0.00e+00	9.13e-07	3.31e-01±0.00e+00	9.13e-07	3.29e-01±0.00e+00	9.13e-07	7.83e-02±0.00e+00	9.13e-07
rbf-SVM	1.10e-01±2.82e-17	9.13e-07	2.87e-01±5.65e-17	9.13e-07	3.01e-01±5.65e-17	9.13e-07	**9.85e-01±5.65e-16**	1.00e+00
poly-SVM	1.12e-01±7.06e-17	9.13e-07	3.62e-01±1.69e-16	9.13e-07	2.97e-01±0.00e+00	9.13e-07	4.00e-01±0.00e+00	9.13e-07
DNM	4.97e-02±9.27e-03	1.24e-06	2.04e-01±1.28e-01	5.70e-03	1.90e-01±1.46e-02	1.01e-06	9.46e-01±2.20e-02	1.00
AISDNM	**3.62e-02±7.67e-03**	-	**1.13e-01±3.51e-02**	-	**1.54e-01±1.97e-02**	-	9.02e-01±3.88e-02	-
Chaos-03								
Model	MSE	p	MAPE	p	MAE	p	R	p
MLP	7.78e-02±2.37e-02	1.24e-06	1.65e-01±4.36e-02	1.01e-06	2.34e-01±4.45e-02	9.13e-07	5.64e-01±1.77e-01	1.24e-06
DT	6.14e-02±0.00e+00	9.13e-07	1.16e-01±0.00e+00	9.13e-07	2.04e-01±2.82e-17	9.13e-07	6.90e-01±2.26e-16	9.13e-07
line-SVM	1.12e-01±4.23e-17	9.13e-07	2.81e-01±1.13e-16	9.13e-07	2.98e-01±5.65e-17	9.13e-07	1.82e-01±2.82e-17	9.13e-07
rbf-SVM	1.04e-01±7.06e-17	9.13e-07	2.78e-01±2.26e-16	9.13e-07	2.86e-01±0.00e+00	9.13e-07	4.11e-01±1.13e-16	9.13e-07
poly-SVM	7.73e-02±2.82e-17	9.13e-07	1.55e-01±5.65e-17	9.13e-07	2.45e-01±1.69e-16	9.13e-07	6.24e-01±2.26e-16	9.13e-07
DNM	**2.56e-02±5.22e-03**	1.00	1.36e-01±6.45e-02	1.95e-05	1.35e-01±1.46e-02	4.84e-01	**9.47e-01±2.68e-02**	1.00
AISDNM	3.15e-02±6.25e-03	-	**4.78e-02±2.90e-02**	-	**1.34e-01±1.36e-02**	-	8.71e-01±3.73e-02	-
Chaos-04								
Model	MSE	p	MAPE	p	MAE	p	R	p
MLP	5.74e-02±1.56e-02	4.03e-06	1.82e-01±4.01e-02	1.51e-06	1.96e-01±3.20e-02	2.04e-06	5.61e-01±1.72e-01	3.33e-06
DT	5.62e-02±2.12e-17	9.13e-07	1.82e-01±8.47e-17	9.13e-07	1.91e-01±1.13e-16	9.13e-07	5.88e-01±0.00e+00	9.13e-07
line-SVM	8.54e-02±4.23e-17	9.13e-07	2.52e-01±1.69e-16	9.13e-07	2.54e-01±1.13e-16	9.13e-07	6.32e-02±1.41e-17	9.13e-07
rbf-SVM	7.49e-02±2.82e-17	9.13e-07	2.17e-01±1.41e-16	9.13e-07	2.38e-01±1.13e-16	9.13e-07	8.00e-01±3.39e-16	2.19e-02
poly-SVM	5.89e-02±3.53e-17	9.13e-07	1.74e-01±8.47e-17	9.13e-07	1.96e-01±2.82e-17	9.13e-07	5.72e-01±2.26e-16	9.13e-07
DNM	8.21e-02±1.57e-02	9.13e-07	3.52e-01±1.54e-01	9.13e-07	2.37e-01±2.86e-02	9.13e-07	7.48e-01±8.91e-02	2.36e-04
AISDNM	**3.13e-02±7.22e-03**	-	**6.86e-02±4.15e-02**	-	**1.37e-01±1.54e-02**	-	**8.27e-01±6.49e-02**	-

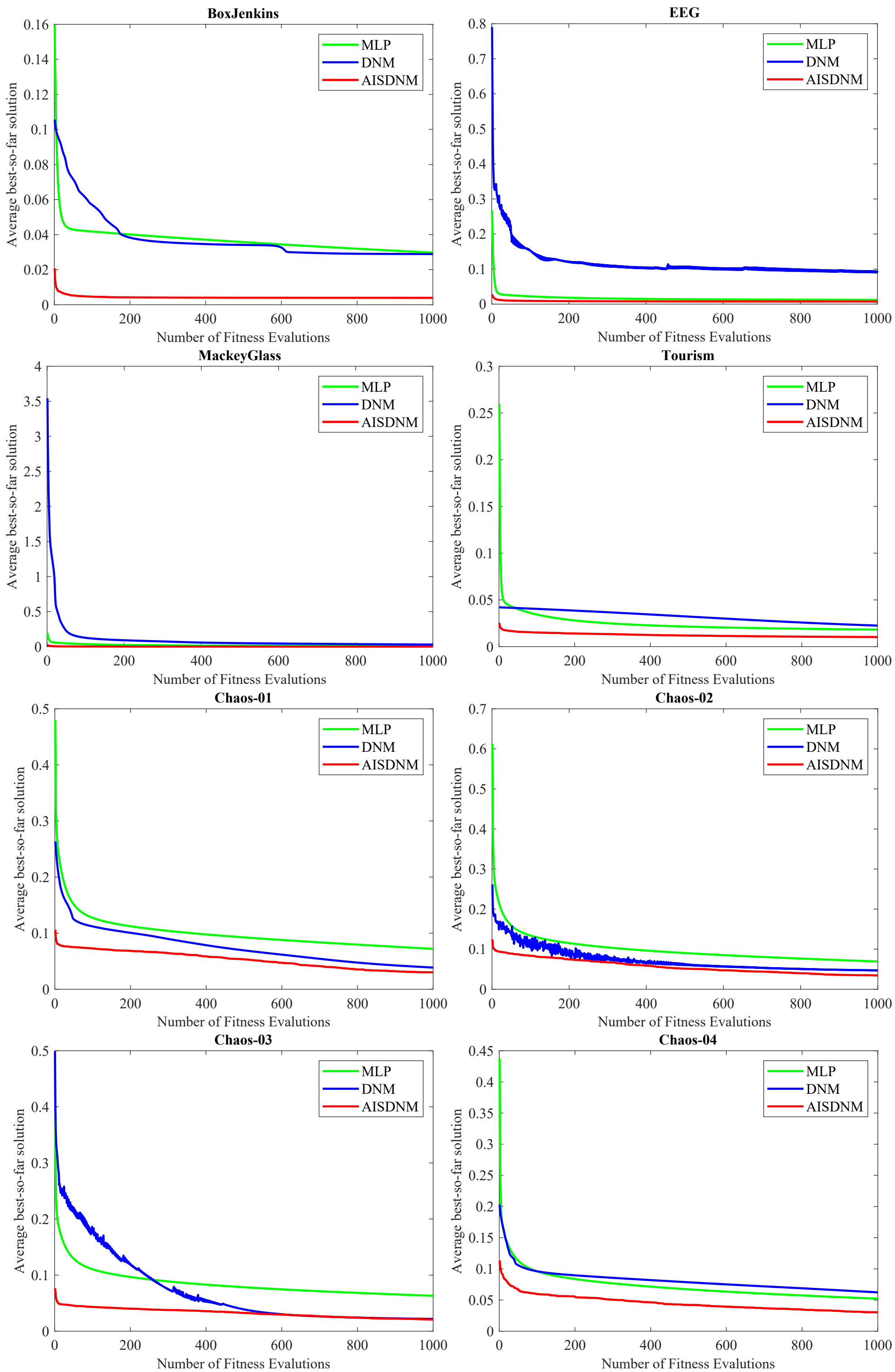

Figure 4.19: The convergence speeds of three models for eight prediction datasets.

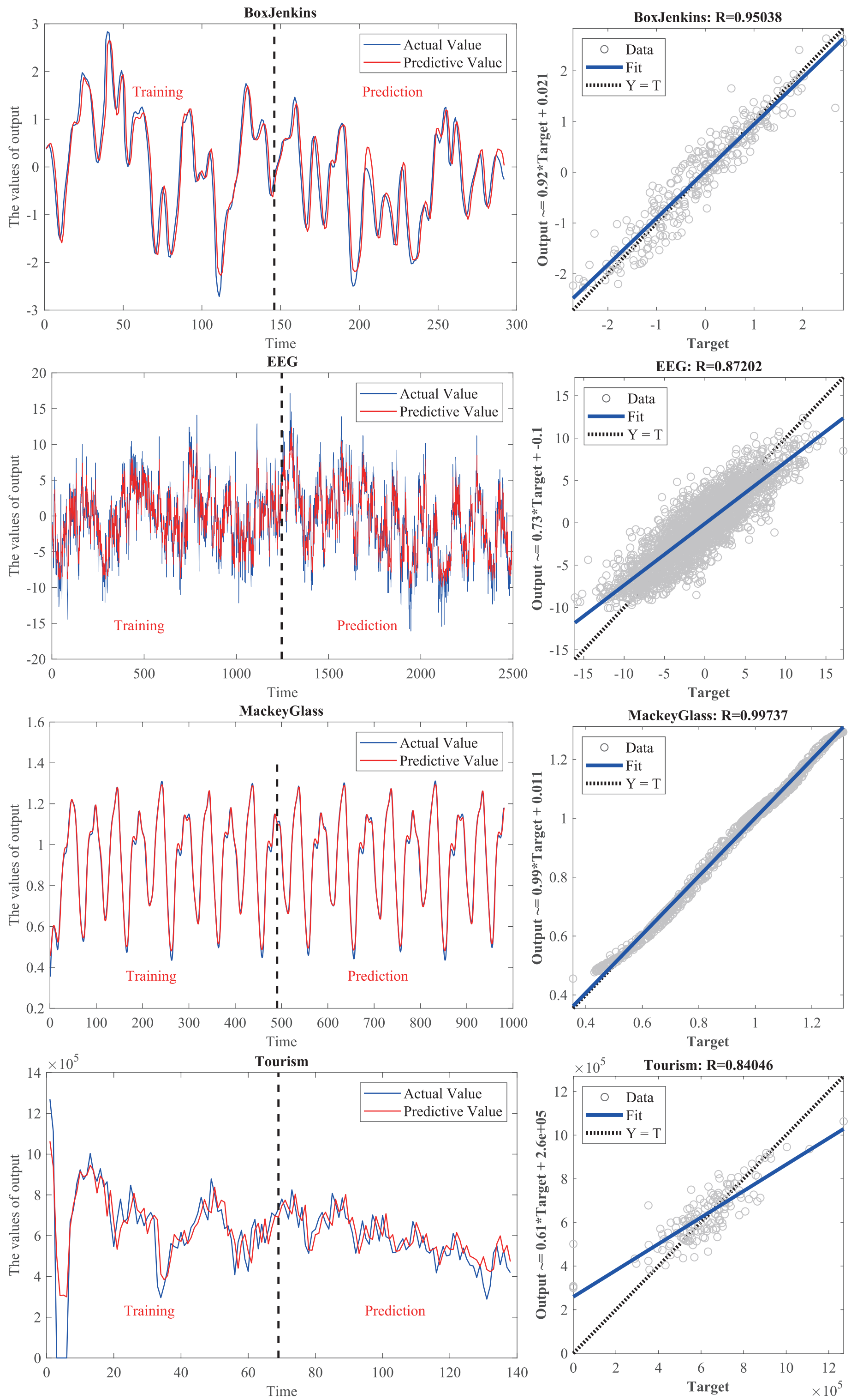

Figure 4.20: The correlation coefficient of prediction of the DNM.

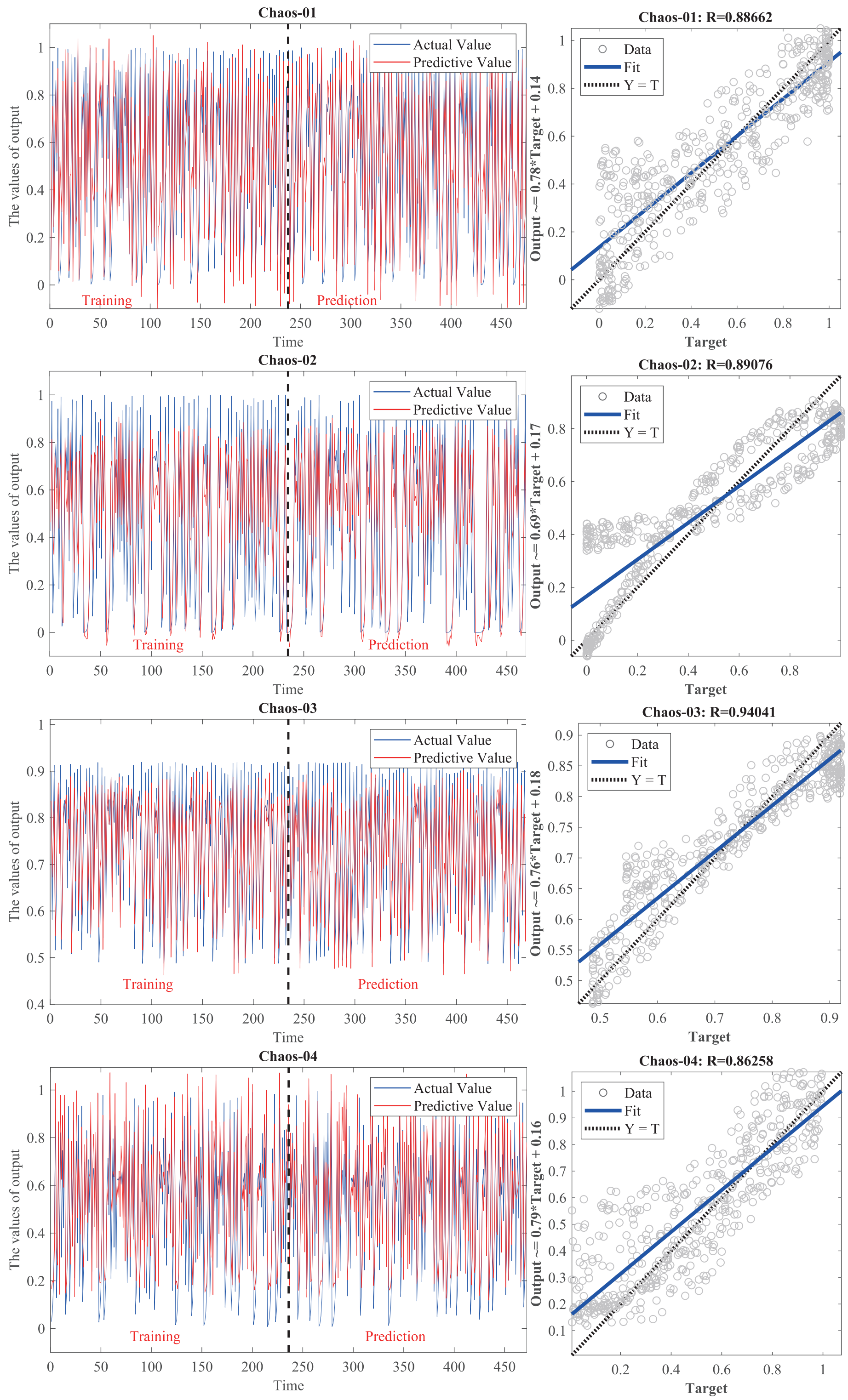

Figure 4.21: The correlation coefficient of prediction of the AISDNM.

Chapter 5

Experimental studies of scale-free cuckoo search

Extensive comparative experimental studies are carried out to test the optimization performance of SFCS. We compare SFCS with seven more popular methods, including the basic CS algorithm, two advanced CS variants, and five metaheuristic algorithms. The comparison among these methods is executed on CEC2013 and CEC2017 benchmark functions.

5.1 Experimental setup

Unless changes are mentioned, the hyperparameters of all the methods are defined as follows: the maximum number of function evaluations is fixed to 10^5; the population size is 50; the search space ranges from -100 to 100; and the search dimension numbers are set to 10, 30 and 50. For the CS and its variants, the probability of the eggs being discarded in each generation P_a is set to 0.25, and the step size of a cuckoo walking in one step α is 0.01. In addition, the user-defined parameters in the other methods are listed in Table 5.1. To achieve the unbiased estimation, all experiments are independently performed 30 times.

Table 5.1: Initial parameters of the five metaheuristic algorithms

Algorithms	Parameters	Values
DE	F	0.7
	CR	0.9
FPA	Pa	0.8
GA	Cop	0.3
	Var	0.1
GSA	G_0	100
	α	20
SMS	α	[0.8, 0.2, 0]
	β	[0.8, 0.4, 0.1]
	ρ	[0.8, 0.3, 0.1]
	H	[0.9, 0.2, 0]

5.2 Benchmark functions

To ensure the results' reliability and generalizability, all the methods are implemented and compared on 58 benchmark functions that have various characteristics. Among them, 28 functions are taken from the CEC'2013, and they can be divided into three categories: 5 unimodal functions ($F_{CEC2013}1 \sim 5$), 15 basic multimodal functions ($F_{CEC2013}6 \sim 20$), and 8 composition functions ($F_{CEC2013}21 \sim 28$). The remaining 30 functions are selected from the CEC'2017, and it includes 3 unimodal functions ($F_{CEC2017}1 \sim 3$), 7 simple multi-modal functions ($F_{CEC2017}4 \sim 10$), 10 hybridity functions ($F_{CEC2017}11 \sim 20$) and 10 composition functions ($F_{CEC2017}21 \sim 30$). Each function is rotated, scaled and shifted to increase the complexity. Complete information of the two benchmark test suites is described in [123] and [124].

5.3 Performance evaluation criteria

In this research, the first criterion is the average (*Average*) and the standard deviation (*Std*) of the final fitness values on 30 independent runs. The second is the nonparametric statistical analysis, which is carried out by the two-sided Wilcoxon rank-sum test to distinguish whether there are significant differences between SFCS and its competitors on a specific function [125]. Specifically, when the null hypothesis is accepted, we summarize these cases as 'w'. A 'l' demonstrates that the compared method obtains superior performance, namely, the SFCS cannot outperform with a significant difference. A 't' implies that the SFCS shows comparable performance to the competitor algorithm. The specific numbers of three categories of statistical results ('$w/t/l$') for all functions in each benchmark special sessions with different values of dimensions are summarized at the bottom of the results tables to make a direct comparison. The last is convergence speed, which is displayed to compare the convergence speed.

5.4 Comparison of the CSs

In this subsection, we compare the SFCS with three CS algorithms, namely, the basic CS algorithm [54], which is used to evaluate the degree of improvement of SFCS, a CS variant algorithm that combines the evolutionary strategy of PSO (PSOCS) [67], and a modified gradient-free optimization CS algorithm (MCS) [70].

In the experiments on 10-dimensional optimization functions, SFCS obtains better results in terms of *Average* and *Std* on 50 (out of 58) benchmark functions from CEC'2013 and CEC'2017, which are listed in Table 5.2 and Table 5.3. For the remaining 8 functions, it can be found that the result of the SFCS is only slightly inferior to the best result. For the sake of further evaluating the performances of the SFCS, the statistical analysis of the two-sided Wilcoxon rank-sum test are provided in Table 5.2 and Table 5.3. It can be found that, compared with the CS, the SFCS is observably superior on 42 functions and comparable on the remaining 16 functions. Compared with PSOCS, the SFCS presents its superiority on all 58 functions. For the MCS algorithm, the SFCS achieves significantly better performances on 52 functions and similar results on 5 functions. The only exception is that SFCS is worse than that of the MCS on $F_{CEC2013}22$. When the dimensional number is set to 30, from Table 5.4 and Table 5.5, it is easy to observe that the CS, PSOCS, MCS and SFCS have the best solutions on 2, 1, 7 and 48 functions, respectively. The statistical analysis suggests that the SFCS is superior to CS on 40 benchmark functions and is similar to it on 17 functions. The exception can be found that the CS outperforms SFCS on $F_{CEC2013}8$. For PSOCS, the SFCS outperforms 57 functions and is outperformed on only 1 function. Compared with the MCS, the SFCS shows satisfactory performance on 46 functions while unsatisfactory on 5 functions. Next, Table 5.6 and Table 5.7 compare the performances of the CSs on the 50-dimensional functions, and the SFCS obtains the best solutions on 42 (out of 58) functions. In summary, the SFCS outperforms the CS, PSOCS and MCS on 44, 56 and 39 functions, while it performs significantly worse on 0, 2 and 10 functions.

In addition, the CSs' convergence curves of 12 randomly selected functions with 10, 30 and 50 dimensions are plotted in Fig. 5.1, which indicates that the speed of the convergence curves of the SFCS ranks first. The convergence speed of the PSOCS and MCS are slower, even compared with the basic CS algorithm, which implies that the SFCS can be regarded as a more effective optimization algorithm. Thus, we can conclude that the SFCS achieves better results on most of the benchmark function with distinct dimensional numbers when compared with the CS, PSOCS and MCS. The scale-free population topology can enhance the search ability of CS prominently.

Table 5.2: Comparison of the CSs on 10-dimensional benchmark functions from CEC'2013

Function	CS Average±Std	PSOCS Average±Std	MCS Average±Std	SFCS Average±Std
$F_{CEC2013}1$	-1.40e+03±5.21e-13	2.14e+03±6.61e+02	-1.40e+03±1.61e-04	**-1.40e+03±3.66e-13**
$F_{CEC2013}2$	6.19e+03±3.69e+03	1.44e+07±5.73e+06	4.40e+05±2.14e+05	**4.81e+03±2.54e+03**
$F_{CEC2013}3$	5.82e+05±3.89e+05	7.38e+09±2.21e+09	1.36e+08±2.23e+08	**1.40e+05±1.92e+05**
$F_{CEC2013}4$	-1.29e+02±3.97e+02	1.43e+04±5.40e+03	2.36e+03±2.04e+03	**-2.40e+02±3.16e+02**
$F_{CEC2013}5$	-1.00e+03±1.04e-08	-4.03e+02±2.33e+02	-1.00e+03±2.96e-03	**-1.00e+03±9.85e-09**
$F_{CEC2013}6$	-9.00e+02±7.14e-02	-6.80e+02±5.10e+01	-8.65e+02±3.28e+01	**-9.00e+02±5.78e-02**
$F_{CEC2013}7$	-7.73e+02±7.78e+00	-6.84e+02±2.29e+01	-7.44e+02±2.52e+01	**-7.76e+02±7.54e+00**
$F_{CEC2013}8$	**-6.80e+02±9.14e-02**	-6.80e+02±7.34e-02	-6.80e+02±8.11e-02	-6.80e+02±6.95e-02
$F_{CEC2013}9$	-5.95e+02±6.72e-01	-5.91e+02±6.72e-01	-5.93e+02±1.46e+00	**-5.95e+02±6.94e-01**
$F_{CEC2013}10$	-5.00e+02±1.68e-02	-1.43e+02±9.75e+01	-4.98e+02±1.12e+00	**-5.00e+02±1.96e-02**
$F_{CEC2013}11$	-3.92e+02±1.86e+00	-2.96e+02±1.24e+01	-3.88e+02±4.79e+00	**-3.94e+02±1.61e+00**
$F_{CEC2013}12$	-2.81e+02±5.79e+00	-1.91e+02±1.47e+01	-2.44e+02±2.28e+01	**-2.82e+02±4.15e+00**
$F_{CEC2013}13$	-1.77e+02±5.82e+00	-9.54e+01±1.27e+01	-1.49e+02±1.56e+01	**-1.79e+02±6.75e+00**
$F_{CEC2013}14$	2.66e+02±6.98e+01	1.51e+03±1.54e+02	1.97e+02±9.44e+01	**1.51e+02±6.72e+01**
$F_{CEC2013}15$	8.67e+02±1.23e+02	1.71e+03±1.61e+02	9.79e+02±3.15e+02	**7.36e+02±1.18e+02**
$F_{CEC2013}16$	2.01e+02±1.26e-01	2.01e+02±2.35e-01	2.01e+02±4.74e-01	**2.01e+02±1.40e-01**
$F_{CEC2013}17$	3.23e+02±2.76e+00	4.84e+02±2.70e+01	3.34e+02±1.19e+01	**3.20e+02±3.93e+00**
$F_{CEC2013}18$	4.33e+02±3.52e+00	5.81e+02±2.76e+01	4.33e+02±1.13e+01	**4.29e+02±5.43e+00**
$F_{CEC2013}19$	5.01e+02±1.73e-01	8.41e+02±2.47e+02	5.02e+02±9.96e-01	**5.01e+02±1.90e-01**
$F_{CEC2013}20$	6.03e+02±2.69e-01	6.04e+02±2.00e-01	6.04e+02±3.47e-01	**6.03e+02±2.19e-01**
$F_{CEC2013}21$	**8.37e+02±4.88e+01**	1.31e+03±5.44e+01	1.10e+03±1.83e+01	8.40e+02±6.21e+01
$F_{CEC2013}22$	1.33e+03±8.85e+01	2.63e+03±1.86e+02	**1.16e+03±1.77e+02**	1.23e+03±1.18e+02
$F_{CEC2013}23$	1.94e+03±1.62e+02	2.76e+03±2.11e+02	2.18e+03±4.17e+02	**1.77e+03±1.82e+02**
$F_{CEC2013}24$	1.18e+03±2.70e+01	1.22e+03±1.44e+01	1.22e+03±5.99e+00	**1.17e+03±3.34e+01**
$F_{CEC2013}25$	**1.27e+03±2.15e+01**	1.31e+03±1.24e+01	1.32e+03±6.08e+00	1.27e+03±2.97e+01
$F_{CEC2013}26$	1.33e+03±8.20e+00	1.40e+03±5.73e+00	1.41e+03±7.35e+01	**1.32e+03±1.00e+01**
$F_{CEC2013}27$	1.68e+03±1.32e+01	2.00e+03±4.25e+01	1.74e+03±9.88e+01	**1.67e+03±2.12e+01**
$F_{CEC2013}28$	**1.54e+03±8.43e+01**	2.30e+03±1.34e+02	2.13e+03±2.01e+02	1.55e+03±8.60e+01
$w/t/l$	19/9 /0	28/0/0	25/2/1	-

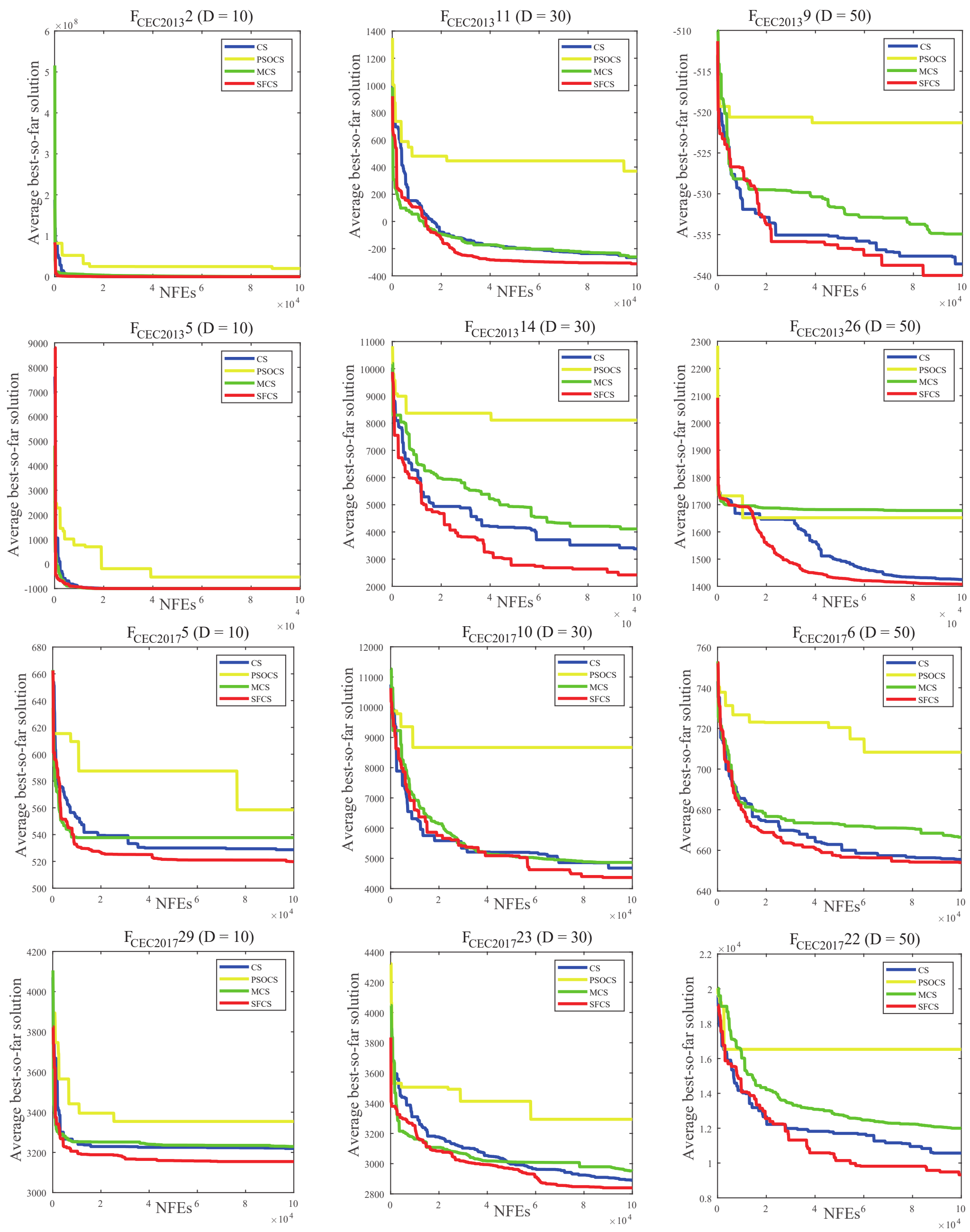

Figure 5.1: Convergence graphs of CS, PSOCS, MCS and SFCS on 12 randomly-selected benchmark functions.

Table 5.3: Comparison of the CSs on 10-dimensional benchmark functions from CEC'2017

Function	CS Average±Std	PSOCS Average±Std	MCS Average±Std	SFCS Average±Std
$F_{CEC2017}1$	1.01e+02±6.11e-01	3.24e+09±8.99e+08	1.18e+03±8.15e+02	**1.01e+02±7.72e-01**
$F_{CEC2017}2$	2.00e+02±8.11e-07	1.46e+09±1.65e+09	2.45e+02±1.65e+02	**2.00e+02±5.34e-07**
$F_{CEC2017}3$	3.00e+02±4.96e-03	1.02e+04±2.29e+03	3.00e+02±9.57e-03	**3.00e+02±2.15e-03**
$F_{CEC2017}4$	4.00e+02±1.73e-01	6.33e+02±6.93e+01	4.05e+02±2.28e+00	**4.00e+02±1.52e-01**
$F_{CEC2017}5$	5.18e+02±5.21e+00	5.71e+02±1.01e+01	5.28e+02±9.13e+00	**5.14e+02±4.40e+00**
$F_{CEC2017}6$	6.05e+02±1.99e+00	6.39e+02±7.18e+00	6.05e+02±4.85e+00	**6.04e+02±1.54e+00**
$F_{CEC2017}7$	7.30e+02±4.20e+00	8.81e+02±2.67e+01	7.46e+02±1.25e+01	**7.25e+02±6.12e+00**
$F_{CEC2017}8$	8.18e+02±4.42e+00	8.69e+02±9.42e+00	8.17e+02±6.79e+00	**8.14e+02±4.58e+00**
$F_{CEC2017}9$	9.25e+02±1.75e+01	1.75e+03±2.37e+02	9.39e+02±5.38e+01	**9.18e+02±1.20e+01**
$F_{CEC2017}10$	1.54e+03±1.16e+02	2.51e+03±1.65e+02	1.78e+03±3.24e+02	**1.46e+03±1.43e+02**
$F_{CEC2017}11$	1.10e+03±1.14e+00	1.42e+03±1.05e+02	1.13e+03±1.25e+01	**1.10e+03±1.30e+00**
$F_{CEC2017}12$	1.58e+03±1.13e+02	8.55e+07±4.12e+07	3.55e+05±2.82e+05	**1.53e+03±1.34e+02**
$F_{CEC2017}13$	1.31e+03±2.45e+00	3.36e+05±3.57e+05	1.11e+04±6.15e+03	**1.31e+03±3.44e+00**
$F_{CEC2017}14$	1.41e+03±3.75e+00	1.80e+03±3.21e+02	1.61e+03±3.11e+02	**1.41e+03±3.19e+00**
$F_{CEC2017}15$	1.50e+03±6.08e-01	4.52e+03±1.63e+03	2.09e+03±1.17e+03	**1.50e+03±7.67e-01**
$F_{CEC2017}16$	1.61e+03±8.17e+00	1.91e+03±8.72e+01	1.83e+03±1.16e+02	**1.60e+03±2.48e+00**
$F_{CEC2017}17$	1.73e+03±3.81e+00	1.83e+03±2.06e+01	1.76e+03±2.92e+01	**1.72e+03±7.25e+00**
$F_{CEC2017}18$	1.81e+03±2.35e+00	8.56e+05±8.84e+05	8.19e+03±6.09e+03	**1.81e+03±2.79e+00**
$F_{CEC2017}19$	1.90e+03±4.33e-01	8.21e+03±5.39e+03	2.62e+03±1.09e+03	**1.90e+03±4.16e-01**
$F_{CEC2017}20$	2.03e+03±5.86e+00	2.15e+03±2.29e+01	2.06e+03±3.59e+01	**2.02e+03±9.20e+00**
$F_{CEC2017}21$	**2.20e+03±1.77e+01**	2.25e+03±1.65e+01	2.32e+03±3.54e+01	2.20e+03±4.75e-01
$F_{CEC2017}22$	2.26e+03±2.93e+01	2.52e+03±9.55e+01	2.30e+03±1.23e+01	**2.26e+03±2.91e+01**
$F_{CEC2017}23$	**2.61e+03±5.06e+01**	2.69e+03±1.61e+01	2.65e+03±1.92e+01	2.62e+03±4.89e+00
$F_{CEC2017}24$	2.51e+03±2.09e+01	2.73e+03±4.90e+01	2.76e+03±8.90e+01	**2.49e+03±2.42e+01**
$F_{CEC2017}25$	2.80e+03±1.27e+02	3.10e+03±5.10e+01	2.93e+03±2.13e+01	**2.77e+03±1.36e+02**
$F_{CEC2017}26$	2.76e+03±8.67e+01	3.38e+03±9.24e+01	3.16e+03±4.90e+02	**2.71e+03±1.06e+02**
$F_{CEC2017}27$	3.10e+03±1.62e+00	3.15e+03±1.47e+01	3.14e+03±4.69e+01	**3.09e+03±1.64e+00**
$F_{CEC2017}28$	**3.02e+03±1.12e+02**	3.41e+03±6.28e+01	3.30e+03±1.34e+02	3.05e+03±1.11e+02
$F_{CEC2017}29$	3.18e+03±1.56e+01	3.32e+03±3.38e+01	3.24e+03±4.12e+01	**3.17e+03±1.22e+01**
$F_{CEC2017}30$	5.26e+03±1.36e+03	4.58e+06±2.32e+06	3.05e+05±5.74e+05	**4.81e+03±1.30e+03**
$w/t/l$	23/7/0	30/0/0	27/3/0	-

Table 5.4: Comparison of the CSs on 30-dimensional benchmark functions from CEC'2013

Function	CS Average±Std	PSOCS Average±Std	MCS Average±Std	SFCS Average±Std
$F_{CEC2013}1$	-1.40e+03±1.27e-02	5.00e+04±4.78e+03	-1.40e+03±9.78e-02	**-1.40e+03±6.78e-03**
$F_{CEC2013}2$	1.28e+07±2.79e+06	6.11e+08±1.06e+08	1.76e+07±5.08e+06	**9.99e+06±2.24e+06**
$F_{CEC2013}3$	2.92e+09±1.14e+09	1.00e+10±0.00e+00	3.84e+09±2.73e+09	**1.88e+09±9.49e+08**
$F_{CEC2013}4$	6.74e+04±9.27e+03	9.15e+04±1.14e+04	**1.73e+04±6.26e+03**	7.19e+04±9.96e+03
$F_{CEC2013}5$	-1.00e+03±1.49e-01	1.25e+04±2.69e+03	-9.99e+02±1.89e-01	**-1.00e+03±1.34e-01**
$F_{CEC2013}6$	-8.61e+02±1.32e+01	6.54e+03±1.13e+03	-8.06e+02±3.32e+01	**-8.70e+02±1.17e+01**
$F_{CEC2013}7$	-6.88e+02±1.06e+01	1.26e+04±1.18e+04	-6.71e+02±3.44e+01	**-6.92e+02±1.51e+01**
$F_{CEC2013}8$	**-6.79e+02±5.53e-02**	-6.79e+02±3.76e-02	-6.79e+02±5.70e-02	-6.79e+02±5.01e-02
$F_{CEC2013}9$	-5.68e+02±1.22e+00	-5.59e+02±1.27e+00	-5.66e+02±2.99e+00	**-5.68e+02±2.09e+00**
$F_{CEC2013}10$	-4.98e+02±2.74e-01	5.72e+03±7.71e+02	-4.87e+02±5.40e+00	**-4.98e+02±2.60e-01**
$F_{CEC2013}11$	-2.58e+02±1.81e+01	4.07e+02±5.99e+01	-2.17e+02±3.42e+01	**-2.77e+02±1.84e+01**
$F_{CEC2013}12$	-7.28e+01±2.64e+01	5.30e+02±7.63e+01	5.48e+01±7.82e+01	**-1.11e+02±2.36e+01**
$F_{CEC2013}13$	6.00e+01±2.24e+01	6.17e+02±5.37e+01	1.37e+02±5.60e+01	**2.81e+01±3.18e+01**
$F_{CEC2013}14$	3.24e+03±2.28e+02	7.77e+03±2.61e+02	3.09e+03±6.20e+02	**2.89e+03±3.61e+02**
$F_{CEC2013}15$	4.65e+03±3.27e+02	7.95e+03±3.01e+02	4.92e+03±7.17e+02	**4.40e+03±2.92e+02**
$F_{CEC2013}16$	2.03e+02±2.87e-01	2.03e+02±4.11e-01	**2.02e+02±7.07e-01**	2.02e+02±2.72e-01
$F_{CEC2013}17$	4.81e+02±1.95e+01	2.02e+03±1.21e+02	6.21e+02±8.19e+01	**4.70e+02±2.22e+01**
$F_{CEC2013}18$	6.34e+02±2.72e+01	2.05e+03±1.24e+02	6.64e+02±4.82e+01	**6.16e+02±2.27e+01**
$F_{CEC2013}19$	5.13e+02±2.06e+00	8.06e+05±2.94e+05	5.28e+02±6.89e+00	**5.12e+02±2.41e+00**
$F_{CEC2013}20$	6.14e+02±2.50e-01	6.15e+02±4.07e-02	6.15e+02±1.92e-01	**6.14e+02±4.44e-01**
$F_{CEC2013}21$	9.43e+02±2.17e+01	4.54e+03±2.09e+02	1.06e+03±7.72e+01	**9.38e+02±2.99e+01**
$F_{CEC2013}22$	4.92e+03±2.62e+02	9.40e+03±2.59e+02	**4.28e+03±9.73e+02**	4.46e+03±3.88e+02
$F_{CEC2013}23$	6.46e+03±2.53e+02	9.38e+03±4.12e+02	7.27e+03±8.03e+02	**6.06e+03±3.95e+02**
$F_{CEC2013}24$	1.28e+03±8.64e+00	1.36e+03±8.41e+00	1.31e+03±2.47e+01	**1.28e+03±7.16e+00**
$F_{CEC2013}25$	1.44e+03±7.17e+00	1.48e+03±5.98e+00	1.45e+03±1.69e+01	**1.43e+03±5.66e+00**
$F_{CEC2013}26$	1.40e+03±2.61e-01	1.47e+03±1.82e+01	1.55e+03±6.99e+01	**1.40e+03±1.94e-01**
$F_{CEC2013}27$	2.53e+03±4.06e+01	2.83e+03±3.66e+01	2.49e+03±1.08e+02	**2.49e+03±5.09e+01**
$F_{CEC2013}28$	1.91e+03±9.14e+01	6.98e+03±5.41e+02	4.11e+03±1.37e+03	**1.88e+03±8.97e+01**
$w/t/l$	19/8/1	28/0/0	23/4/1	-

Table 5.5: Comparison of the CSs on 30-dimensional benchmark functions from CEC'2017

Function	CS	PSOCS	MCS	SFCS
	Average±Std	Average±Std	Average±Std	Average±Std
$F_{CEC2017}1$	4.80e+05±1.61e+05	1.00e+10±0.00e+00	4.93e+05±1.66e+05	**3.15e+05±1.40e+05**
$F_{CEC2017}2$	6.36e+17±9.25e+17	**1.00e+10±0.00e+00**	1.61e+14±6.46e+14	1.83e+17±3.68e+17
$F_{CEC2017}3$	8.86e+04±1.27e+04	1.31e+05±2.05e+04	**7.20e+03±3.39e+03**	8.10e+04±9.55e+03
$F_{CEC2017}4$	4.85e+02±1.53e+01	1.22e+04±2.43e+03	5.19e+02±1.48e+01	**4.81e+02±1.94e+01**
$F_{CEC2017}5$	6.64e+02±1.73e+01	9.69e+02±2.05e+01	6.76e+02±3.30e+01	**6.52e+02±1.88e+01**
$F_{CEC2017}6$	6.48e+02±6.96e+00	6.93e+02±4.88e+00	**6.43e+02±6.78e+00**	6.43e+02±5.18e+00
$F_{CEC2017}7$	8.83e+02±1.78e+01	2.37e+03±1.31e+02	1.04e+03±7.06e+01	**8.76e+02±2.47e+01**
$F_{CEC2017}8$	9.50e+02±2.12e+01	1.22e+03±1.89e+01	**9.24e+02±2.43e+01**	9.31e+02±1.87e+01
$F_{CEC2017}9$	4.85e+03±7.82e+02	1.76e+04±1.94e+03	**3.63e+03±8.99e+02**	4.57e+03±9.99e+02
$F_{CEC2017}10$	4.86e+03±2.27e+02	8.67e+03±3.89e+02	5.12e+03±6.15e+02	**4.46e+03±3.64e+02**
$F_{CEC2017}11$	1.24e+03±1.67e+01	9.27e+03±1.96e+03	1.26e+03±3.90e+01	**1.22e+03±2.03e+01**
$F_{CEC2017}12$	5.46e+05±2.61e+05	8.39e+09±1.16e+09	4.80e+06±2.64e+06	**3.49e+05±1.44e+05**
$F_{CEC2017}13$	1.18e+04±3.30e+03	3.35e+09±9.92e+08	5.07e+04±1.91e+04	**1.13e+04±2.78e+03**
$F_{CEC2017}14$	1.65e+03±5.63e+01	1.31e+06±7.37e+05	2.42e+04±1.65e+04	**1.63e+03±3.76e+01**
$F_{CEC2017}15$	2.61e+03±3.52e+02	4.77e+08±2.05e+08	2.79e+04±1.83e+04	**2.45e+03±2.22e+02**
$F_{CEC2017}16$	2.76e+03±1.53e+02	4.92e+03±3.27e+02	3.05e+03±2.96e+02	**2.64e+03±1.27e+02**
$F_{CEC2017}17$	2.08e+03±7.21e+01	3.28e+03±2.09e+02	2.49e+03±1.84e+02	**2.02e+03±1.06e+02**
$F_{CEC2017}18$	7.05e+04±2.60e+04	2.15e+07±8.88e+06	6.43e+05±5.53e+05	**6.84e+04±2.37e+04**
$F_{CEC2017}19$	**2.10e+03±6.46e+01**	6.21e+08±2.14e+08	4.93e+04±3.31e+04	2.12e+03±8.17e+01
$F_{CEC2017}20$	2.45e+03±8.00e+01	2.98e+03±8.55e+01	2.62e+03±1.80e+02	**2.36e+03±8.58e+01**
$F_{CEC2017}21$	2.46e+03±1.85e+01	2.72e+03±2.88e+01	2.47e+03±3.15e+01	**2.44e+03±2.02e+01**
$F_{CEC2017}22$	2.71e+03±7.49e+02	8.82e+03±6.06e+02	3.53e+03±2.09e+03	**2.36e+03±2.10e+01**
$F_{CEC2017}23$	2.90e+03±4.29e+01	3.32e+03±5.00e+01	3.00e+03±8.19e+01	**2.87e+03±4.05e+01**
$F_{CEC2017}24$	3.03e+03±2.92e+01	3.54e+03±6.13e+01	3.14e+03±8.01e+01	**2.99e+03±2.94e+01**
$F_{CEC2017}25$	2.89e+03±1.25e+00	7.27e+03±7.98e+02	2.92e+03±1.96e+01	**2.89e+03±8.98e-01**
$F_{CEC2017}26$	3.93e+03±5.76e+02	1.02e+04±5.46e+02	6.18e+03±1.71e+03	**3.50e+03±5.76e+02**
$F_{CEC2017}27$	3.34e+03±1.90e+01	3.96e+03±1.06e+02	3.39e+03±8.72e+01	**3.31e+03±2.22e+01**
$F_{CEC2017}28$	3.22e+03±1.36e+01	7.15e+03±5.02e+02	3.27e+03±2.92e+01	**3.22e+03±1.06e+01**
$F_{CEC2017}29$	4.15e+03±1.14e+02	6.00e+03±3.52e+02	4.07e+03±2.31e+02	**3.99e+03±1.39e+02**
$F_{CEC2017}30$	8.90e+04±4.14e+04	4.59e+08±1.38e+08	4.60e+05±1.85e+05	**6.81e+04±2.61e+04**
$w/t/l$	21/9/0	29/0/1	23/3/4	-

Table 5.6: Comparison of the CSs on 50-dimensional benchmark functions from CEC'2013

Function	CS Average±Std	PSOCS Average±Std	MCS Average±Std	SFCS Average±Std
$F_{CEC2013}1$	-1.39e+03±4.08e+00	1.08e+05±7.35e+03	-1.39e+03±1.75e+00	**-1.39e+03±2.19e+00**
$F_{CEC2013}2$	3.62e+07±5.03e+06	1.89e+09±2.41e+08	4.13e+07±1.25e+07	**2.92e+07±5.96e+06**
$F_{CEC2013}3$	1.83e+10±4.08e+09	**1.00e+10±0.00e+00**	1.49e+10±7.75e+09	1.51e+10±2.63e+09
$F_{CEC2013}4$	1.24e+05±1.58e+04	1.62e+05±1.81e+04	**4.36e+04±1.14e+04**	1.22e+05±1.55e+04
$F_{CEC2013}5$	-9.76e+02±7.43e+00	3.02e+04±4.66e+03	-9.79e+02±6.73e+00	**-9.82e+02±3.84e+00**
$F_{CEC2013}6$	-8.43e+02±1.26e+01	1.29e+04±1.80e+03	-7.27e+02±5.71e+01	**-8.45e+02±8.96e+00**
$F_{CEC2013}7$	-6.63e+02±1.06e+01	1.00e+04±6.80e+03	-6.50e+02±5.71e+01	**-6.67e+02±1.37e+01**
$F_{CEC2013}8$	**-6.79e+02±5.01e-02**	-6.79e+02±3.92e-02	-6.79e+02±4.11e-02	-6.79e+02±2.55e-02
$F_{CEC2013}9$	-5.36e+02±1.71e+00	-5.24e+02±1.13e+00	-5.36e+02±4.72e+00	**-5.39e+02±1.81e+00**
$F_{CEC2013}10$	-4.53e+02±1.65e+01	1.33e+04±1.01e+03	-3.70e+02±3.20e+01	**-4.64e+02±1.06e+01**
$F_{CEC2013}11$	-7.12e+01±3.02e+01	1.31e+03±1.01e+02	3.66e+00±5.20e+01	**-1.03e+02±3.29e+01**
$F_{CEC2013}12$	1.77e+02±3.36e+01	1.33e+03±1.00e+02	3.50e+02±1.16e+02	**1.37e+02±5.32e+01**
$F_{CEC2013}13$	3.56e+02±3.76e+01	1.44e+03±9.21e+01	4.81e+02±9.84e+01	**3.11e+02±3.83e+01**
$F_{CEC2013}14$	7.87e+03±3.42e+02	1.46e+04±3.54e+02	**6.15e+03±8.32e+02**	7.31e+03±3.71e+02
$F_{CEC2013}15$	1.05e+04±4.06e+02	1.52e+04±4.02e+02	**9.82e+03±1.16e+03**	1.01e+04±3.48e+02
$F_{CEC2013}16$	2.04e+02±3.09e-01	2.04e+02±5.30e-01	2.03e+02±7.86e-01	**2.03e+02±4.64e-01**
$F_{CEC2013}17$	7.53e+02±3.69e+01	4.09e+03±1.30e+02	1.15e+03±1.30e+02	**7.22e+02±3.75e+01**
$F_{CEC2013}18$	9.32e+02±4.40e+01	4.12e+03±1.91e+02	1.09e+03±1.19e+02	**9.29e+02±3.90e+01**
$F_{CEC2013}19$	5.49e+02±7.48e+00	4.00e+06±1.05e+06	5.64e+02±1.16e+01	**5.43e+02±8.37e+00**
$F_{CEC2013}20$	**6.24e+02±3.29e-01**	6.25e+02±6.34e-03	6.24e+02±2.83e-01	6.24e+02±2.72e-01
$F_{CEC2013}21$	1.26e+03±1.10e+02	9.14e+03±3.85e+02	1.55e+03±2.92e+02	**1.17e+03±1.21e+02**
$F_{CEC2013}22$	1.06e+04±3.97e+02	1.66e+04±4.39e+02	**9.38e+03±1.40e+03**	9.77e+03±4.99e+02
$F_{CEC2013}23$	1.28e+04±4.46e+02	1.70e+04±3.60e+02	1.34e+04±1.38e+03	**1.21e+04±6.81e+02**
$F_{CEC2013}24$	1.36e+03±1.03e+01	1.54e+03±3.01e+01	1.39e+03±5.96e+01	**1.35e+03±1.15e+01**
$F_{CEC2013}25$	1.58e+03±7.91e+00	1.65e+03±1.15e+01	1.58e+03±2.77e+01	**1.57e+03±1.37e+01**
$F_{CEC2013}26$	1.42e+03±9.54e+00	1.63e+03±4.40e+01	1.66e+03±4.98e+01	**1.41e+03±2.90e+00**
$F_{CEC2013}27$	3.45e+03±5.68e+01	3.97e+03±6.97e+01	3.44e+03±1.38e+02	**3.37e+03±7.24e+01**
$F_{CEC2013}28$	1.90e+03±3.29e+01	1.22e+04±7.18e+02	4.99e+03±2.50e+03	**1.89e+03±2.04e+01**
$w/t/l$	20/8/0	27/0/1	20/4/4	-

Table 5.7: Comparison of the CSs on 50-dimensional benchmark functions from CEC'2017

Function	CS	PSOCS	MCS	SFCS
	Average±Std	Average±Std	Average±Std	Average±Std
$F_{CEC2017}1$	3.30e+07±1.22e+07	1.00e+10±0.00e+00	**2.09e+07±5.45e+06**	2.67e+07±1.32e+07
$F_{CEC2017}2$	4.06e+44±8.60e+44	**1.00e+10±0.00e+00**	1.61e+34±4.40e+34	1.82e+42±7.41e+42
$F_{CEC2017}3$	2.19e+05±2.75e+04	2.75e+05±3.13e+04	**6.19e+04±1.54e+04**	2.24e+05±3.28e+04
$F_{CEC2017}4$	6.18e+02±3.34e+01	4.45e+04±4.78e+03	6.39e+02±5.32e+01	**5.97e+02±4.71e+01**
$F_{CEC2017}5$	8.42e+02±3.01e+01	1.40e+03±3.83e+01	**8.07e+02±3.61e+01**	8.12e+02±3.95e+01
$F_{CEC2017}6$	6.66e+02±4.79e+00	7.14e+02±4.12e+00	**6.56e+02±4.32e+00**	6.62e+02±5.50e+00
$F_{CEC2017}7$	1.17e+03±3.81e+01	4.46e+03±1.89e+02	1.52e+03±1.21e+02	**1.12e+03±4.01e+01**
$F_{CEC2017}8$	1.15e+03±2.34e+01	1.70e+03±3.36e+01	1.11e+03±4.82e+01	**1.11e+03±2.66e+01**
$F_{CEC2017}9$	1.91e+04±3.13e+03	5.77e+04±4.87e+03	**1.29e+04±2.79e+03**	1.70e+04±2.51e+03
$F_{CEC2017}10$	8.98e+03±3.46e+02	1.52e+04±3.49e+02	**8.04e+03±9.99e+02**	8.57e+03±3.79e+02
$F_{CEC2017}11$	1.67e+03±8.98e+01	2.96e+04±4.80e+03	**1.44e+03±7.72e+01**	1.52e+03±6.99e+01
$F_{CEC2017}12$	1.48e+07±4.47e+06	1.00e+10±0.00e+00	2.89e+07±1.67e+07	**9.59e+06±3.52e+06**
$F_{CEC2017}13$	9.27e+04±3.67e+04	1.00e+10±0.00e+00	1.11e+05±4.83e+04	**6.71e+04±2.42e+04**
$F_{CEC2017}14$	**3.99e+04±2.26e+04**	1.70e+07±5.98e+06	4.47e+05±2.52e+05	4.20e+04±1.97e+04
$F_{CEC2017}15$	1.13e+04±3.59e+03	5.78e+09±1.07e+09	3.14e+04±1.76e+04	**1.08e+04±3.33e+03**
$F_{CEC2017}16$	3.77e+03±2.15e+02	8.05e+03±4.68e+02	3.65e+03±4.49e+02	**3.41e+03±2.85e+02**
$F_{CEC2017}17$	3.17e+03±1.29e+02	9.72e+03±2.33e+03	3.55e+03±3.43e+02	**2.97e+03±1.52e+02**
$F_{CEC2017}18$	1.65e+06±4.96e+05	9.46e+07±2.94e+07	3.28e+06±1.69e+06	**1.08e+06±5.23e+05**
$F_{CEC2017}19$	1.68e+04±4.03e+03	2.59e+09±5.61e+08	2.02e+05±9.47e+04	**1.63e+04±4.26e+03**
$F_{CEC2017}20$	3.22e+03±1.23e+02	4.31e+03±1.29e+02	3.35e+03±2.86e+02	**3.11e+03±1.60e+02**
$F_{CEC2017}21$	2.67e+03±2.77e+01	3.20e+03±3.38e+01	2.68e+03±6.50e+01	**2.62e+03±3.74e+01**
$F_{CEC2017}22$	1.08e+04±3.72e+02	1.68e+04±3.87e+02	1.04e+04±9.02e+02	**1.01e+04±7.78e+02**
$F_{CEC2017}23$	3.37e+03±6.83e+01	4.18e+03±1.05e+02	3.51e+03±1.16e+02	**3.29e+03±6.41e+01**
$F_{CEC2017}24$	3.42e+03±6.39e+01	4.50e+03±1.05e+02	3.53e+03±1.03e+02	**3.37e+03±5.58e+01**
$F_{CEC2017}25$	3.11e+03±2.54e+01	2.73e+04±3.52e+03	3.19e+03±2.98e+01	**3.09e+03±2.69e+01**
$F_{CEC2017}26$	8.36e+03±8.96e+02	1.97e+04±8.95e+02	1.07e+04±1.04e+03	**8.21e+03±1.04e+03**
$F_{CEC2017}27$	4.20e+03±9.78e+01	5.96e+03±2.69e+02	4.35e+03±2.92e+02	**4.05e+03±1.28e+02**
$F_{CEC2017}28$	3.42e+03±4.41e+01	1.46e+04±8.72e+02	3.51e+03±5.38e+01	**3.41e+03±3.67e+01**
$F_{CEC2017}29$	5.24e+03±2.58e+02	1.67e+04±4.26e+03	5.13e+03±4.25e+02	**5.05e+03±1.99e+02**
$F_{CEC2017}30$	1.51e+07±2.62e+06	4.02e+09±8.48e+08	1.52e+07±2.65e+06	**1.26e+07±1.88e+06**
$w/t/l$	24/6/0	29/0/1	19/5/6	-

5.5 Comparison of the SFCS with five metaheuristic algorithms

To further illustrate the effectiveness of the SFCS, five representative metaheuristic algorithms are chosen for comparison with the SFCS, namely, DE [126], the FPA, GA [127], the GSA and the SMS [128]. For the DE, the corresponding hyper-parameters refer to the analysis in [129]. The hyper-parameters of the FPA are set according to the analysis in [130]. For the GA, the corresponding hyper-parameters are consistent with the settings in [131]. The analysis in [132] suggests that the G_0 of GSA is fixed to 100 and α is 20. The hyper-parameters of the SMS are set according to the analysis in [133].

Table 5.8 and Table 5.9 show the optimization results obtained by these algorithms. Upon optimizing the 10-dimensional functions, DE, FPA, GA, GSA, SMS and SFCS achieve the best solutions on 27, 5, 2, 9, 0 and 15 functions. In addition, Wilcoxon rank-sum test is adopted. Specifically, the SFCS outperforms DE on 23 functions and has a comparable performance on 7 functions. In this case, the DE obtains better performance than SFCS. For FPA, the performance of the SFCS is better on 45 functions but worse on the 5 benchmark functions. Compared with the GA, the SFCS has better performances on 53 functions and performs similarly on 2 functions. The SFCS outperforms the GSA on 45 functions obviously and is outperformed by the GSA on 10 functions. Moreover, the SFCS performs better than the SMS on 56 functions and worse on 1 function. Upon optimizing 30-dimensional functions, as shown in Table 5.10 and Table 5.11, the SFCS has advantages over the other algorithms on number 18, 5, 4, 8, 2 and 21. Specifically, the SFCS obviously outperforms the DE on 34 functions and has a comparable performance on 7 functions. The SFCS is distinctly superior to the FPA on 50 functions and worse on the remaining 8 functions. Compared with the GA, the SFCS achieves more satisfactory performance on 49 functions but unsatisfactory on 4 functions. Moreover, the SFCS evidently performs better than GSA on 45 functions but is outperformed on the remaining 11 functions. For SMS, the SFCS is obviously superior on 53 functions but inferior on 4 functions. From Table 5.12 and Table 5.13, it is obvious that the DE, FPA, GA, GSA, SMS and SFCS obtain the best solutions on 18, 4, 3, 9, 2 and 22 functions separately with 50 dimensions. The SFCS solutions are distinctly superior to those of the DE, FPA, GA, GSA and SMS on 35, 49, 51, 39 and 51 functions, respectively.

Hence, these results suggest that the SFCS is an outstanding algorithm among numerous metaheuristic algorithms for various functions with different dimensions. Fig. 5.2 depicts the convergence curves of each algorithm. It is obvious that the other five metaheuristic algorithms converge fast in the early phase, but they fall into the local minima in a later phase. The SFCS continues to converge consistently in the whole search process and finally has the best solutions in contrast with other algorithms. Generally, the results verify that the SFCS not only has structural simplicity but also presents outstanding performance and high computational efficiency due to its scale-free population topology.

5.6 Discussion

The previous results present convincing proof that the scale-free population topology plays an irreplaceable guiding role in the process of searching for solutions. It successfully enables the CS to achieve a better agreement between exploitation and exploration. Compared with the basic CS algorithm, two CS variants, and five metaheuristic optimization algorithms, the SFCS showed promising and comparable performance. Additionally, further analysis of SFCS algorithm is carried out. First, the influence of the parameter M_0 on the performance of the SFCS is detected and analyzed. Then, we attempt to apply SFCS architecture design to resolve real-world tasks. The comparisons are also provided with the statistical results.

5.6.1 Parameter sensibility

There is an important parameter named M_0 in the scale-free population topology, which indicates the number of fully attached nodes in the initialization phase. It

Table 5.8: Comparison of SFCS with the five metaheuristic algorithms on 10-dimensional benchmark functions from CEC'2013

Function	DE	FPA	GA	GSA	SMS	SFCS
	Average±Std	Average±Std	Average±Std	Average±Std	Average±Std	Average±Std
$F_{CEC2013}1$	-1.40e+03±0.00e+00	-1.40e+03±2.56e-05	-1.36e+03±2.40e+01	**-1.40e+03±0.00e+00**	8.23e+03±2.31e+03	-1.40e+03±2.39e-13
$F_{CEC2013}2$	**-1.30e+03±2.64e-09**	-7.49e+02±3.64e+02	1.57e+07±1.09e+07	2.02e+06±5.95e+05	4.89e+07±2.54e+07	4.29e+03±2.54e+03
$F_{CEC2013}3$	**-1.20e+03±8.39e-02**	8.07e+06±5.32e+06	4.84e+09±3.99e+09	1.58e+08±2.22e+08	1.00e+08±0.00e+00	1.57e+05±1.78e+05
$F_{CEC2013}4$	**-1.10e+03±2.30e-11**	-6.55e+02±2.02e+02	3.16e+04±1.29e+04	1.68e+04±2.26e+03	1.80e+04±2.81e+03	-1.26e+02±3.78e+02
$F_{CEC2013}5$	**-1.00e+03±0.00e+00**	-1.00e+03±1.70e-03	-9.85e+02±9.44e+00	-1.00e+03±2.58e-05	4.25e+03±2.22e+03	-1.00e+03±1.23e-08
$F_{CEC2013}6$	-8.94e+02±4.03e+00	-9.00e+02±8.28e-02	-8.42e+02±4.59e+01	-8.52e+02±6.99e+00	-1.65e+02±2.50e+02	**-9.00e+02±1.05e-01**
$F_{CEC2013}7$	**-8.00e+02±4.91e-03**	-7.73e+02±8.91e+00	-6.84e+02±6.86e+01	-7.80e+02±1.16e+01	-7.88e+01±1.39e+03	-7.74e+02±8.04e+00
$F_{CEC2013}8$	**-6.80e+02±7.98e-02**	-6.80e+02±6.76e-02	-6.80e+02±8.54e-02	-6.80e+02±8.64e-02	-6.80e+02±6.88e-02	-6.80e+02±6.05e-02
$F_{CEC2013}9$	-5.92e+02±2.46e+00	-5.94e+02±6.54e-01	-5.93e+02±8.49e-01	**-5.96e+02±1.40e+00**	-5.88e+02±1.07e+00	-5.95e+02±8.43e-01
$F_{CEC2013}10$	-5.00e+02±1.89e-01	-5.00e+02±1.85e-02	-4.30e+02±3.42e+01	**-5.00e+02±6.59e-03**	5.17e+02±2.70e+02	-5.00e+02±2.10e-02
$F_{CEC2013}11$	-3.93e+02±7.12e+00	-3.84e+02±2.76e+00	-3.89e+02±3.60e+00	-3.76e+02±6.54e+00	-2.04e+02±3.16e+01	**-3.94e+02±1.88e+00**
$F_{CEC2013}12$	-2.77e+02±1.04e+01	-2.82e+02±4.31e+00	-2.30e+02±2.04e+01	-2.78e+02±7.18e+00	-1.29e+02±2.02e+01	**-2.83e+02±5.63e+00**
$F_{CEC2013}13$	-1.79e+02±1.01e+01	-1.76e+02±5.06e+00	-1.22e+02±2.01e+01	-1.54e+02±1.11e+01	-3.00e+01±1.67e+01	**-1.81e+02±6.49e+00**
$F_{CEC2013}14$	8.80e+02±2.78e+02	4.75e+02±1.11e+02	**3.71e+01±6.49e+01**	6.83e+02±2.31e+02	2.38e+03±2.24e+02	1.62e+02±7.83e+01
$F_{CEC2013}15$	1.59e+03±2.26e+02	8.03e+02±1.01e+02	1.30e+03±2.15e+02	**5.88e+02±2.13e+02**	1.83e+03±2.94e+02	6.86e+02±1.82e+02
$F_{CEC2013}16$	2.01e+02±1.73e-01	2.01e+02±1.95e-01	2.01e+02±2.66e-01	**2.00e+02±1.74e-02**	2.00e+02±2.99e-01	2.01e+02±1.31e-01
$F_{CEC2013}17$	3.24e+02±8.61e+00	3.30e+02±5.13e+00	3.26e+02±4.68e+00	**3.12e+02±1.15e+00**	4.35e+02±1.23e+01	3.21e+02±3.31e+00
$F_{CEC2013}18$	4.39e+02±5.10e+00	4.34e+02±4.13e+00	4.85e+02±1.56e+01	**4.13e+02±1.29e+00**	5.29e+02±9.43e+00	4.27e+02±3.97e+00
$F_{CEC2013}19$	5.01e+02±7.95e-01	5.01e+02±1.93e-01	5.03e+02±1.26e+00	5.01e+02±3.11e-01	1.58e+04±8.93e+03	**5.01e+02±2.05e-01**
$F_{CEC2013}20$	**6.03e+02±3.12e-01**	6.03e+02±1.92e-01	6.04e+02±3.64e-01	6.04e+02±4.43e-01	6.05e+02±1.86e-01	6.03e+02±3.87e-01
$F_{CEC2013}21$	1.09e+03±5.08e+01	8.77e+02±8.12e+01	1.10e+03±4.62e+01	1.10e+03±4.63e-13	1.33e+03±4.12e+01	**8.30e+02±7.02e+01**
$F_{CEC2013}22$	2.02e+03±2.90e+02	1.55e+03±1.25e+02	**1.02e+03±8.88e+01**	2.77e+03±2.43e+02	3.69e+03±2.32e+02	1.20e+03±1.19e+02
$F_{CEC2013}23$	2.55e+03±1.76e+02	1.95e+03±1.24e+02	2.31e+03±2.23e+02	2.13e+03±2.49e+02	3.40e+03±1.93e+02	**1.81e+03±1.84e+02**
$F_{CEC2013}24$	1.21e+03±1.01e+01	1.17e+03±2.63e+01	1.19e+03±3.72e+01	1.23e+03±4.67e+00	1.25e+03±9.75e+00	**1.16e+03±3.09e+01**
$F_{CEC2013}25$	1.32e+03±3.09e+00	**1.26e+03±1.71e+01**	1.28e+03±3.20e+01	1.32e+03±8.33e+00	1.35e+03±4.04e+00	1.26e+03±2.98e+01
$F_{CEC2013}26$	1.32e+03±7.58e+00	1.33e+03±9.64e+00	1.39e+03±1.99e+01	1.51e+03±4.36e+01	1.51e+03±4.27e+01	**1.32e+03±7.79e+00**
$F_{CEC2013}27$	**1.60e+03±2.29e-03**	1.70e+03±1.37e+01	1.92e+03±6.59e+01	1.70e+03±4.94e-10	2.13e+03±4.19e+01	1.66e+03±3.53e+01
$F_{CEC2013}28$	1.69e+03±5.07e+01	**1.53e+03±6.96e+01**	2.07e+03±2.46e+02	2.01e+03±1.14e+02	2.59e+03±3.85e+01	1.54e+03±8.13e+01
$w/t/l$	15/5/8	22/4/2	25/1/2	19/1/8	26/1/1	-

Table 5.9: Comparison of SFCS with the five metaheuristic algorithms on 10-dimensional benchmark functions from CEC'2017

Function	DE	FPA	GA	GSA	SMS	SFCS
	Average±Std	Average±Std	Average±Std	Average±Std	Average±Std	Average±Std
$F_{CEC2017}1$	**1.00e+02±0.00e+00**	3.74e+03±2.07e+03	6.38e+07±4.79e+07	4.53e+02±5.59e+02	1.00e+08±0.00e+00	1.01e+02±5.56e-01
$F_{CEC2017}2$	**2.00e+02±0.00e+00**	2.00e+02±4.84e-04	4.01e+08±8.19e+08	2.58e+02±3.91e+01	1.00e+08±0.00e+00	2.00e+02±4.01e-07
$F_{CEC2017}3$	**3.00e+02±0.00e+00**	3.00e+02±1.61e-04	2.60e+04±1.34e+04	5.34e+03±1.38e+03	1.59e+04±2.29e+03	3.00e+02±2.07e-03
$F_{CEC2017}4$	**4.00e+02±0.00e+00**	4.00e+02±1.09e-01	4.43e+02±4.32e+01	4.05e+02±3.81e-01	1.66e+03±4.85e+02	4.00e+02±1.52e-01
$F_{CEC2017}5$	**5.14e+02±8.76e+00**	5.18e+02±3.86e+00	5.21e+02±6.10e+00	5.47e+02±8.07e+00	6.15e+02±1.90e+01	5.15e+02±4.71e+00
$F_{CEC2017}6$	**6.00e+02±8.45e-10**	6.10e+02±2.92e+00	6.03e+02±1.25e+00	6.06e+02±6.82e+00	6.64e+02±9.15e+00	6.05e+02±2.20e+00
$F_{CEC2017}7$	7.33e+02±9.22e+00	7.30e+02±4.16e+00	7.53e+02±8.60e+00	**7.13e+02±1.36e+00**	8.09e+02±1.02e+01	7.26e+02±5.50e+00
$F_{CEC2017}8$	8.16e+02±9.39e+00	**8.14e+02±3.28e+00**	8.20e+02±5.70e+00	8.22e+02±4.85e+00	8.86e+02±1.28e+01	8.15e+02±4.42e+00
$F_{CEC2017}9$	9.00e+02±0.00e+00	9.17e+02±9.52e+00	1.01e+03±7.92e+01	**9.00e+02±0.00e+00**	1.70e+03±2.18e+02	9.21e+02±1.34e+01
$F_{CEC2017}10$	2.16e+03±2.71e+02	1.59e+03±1.17e+02	1.63e+03±1.86e+02	2.73e+03±2.80e+02	3.36e+03±2.48e+02	**1.47e+03±1.34e+02**
$F_{CEC2017}11$	**1.10e+03±8.64e-01**	1.11e+03±1.27e+00	1.69e+03±5.69e+02	1.13e+03±7.65e+00	7.66e+03±4.48e+03	1.10e+03±1.39e+00
$F_{CEC2017}12$	**1.20e+03±2.81e+00**	3.18e+03±5.09e+02	8.81e+06±7.77e+06	2.39e+05±2.34e+05	9.87e+07±6.96e+06	1.53e+03±1.24e+02
$F_{CEC2017}13$	**1.30e+03±2.57e+00**	1.36e+03±1.23e+01	1.51e+05±1.57e+05	1.16e+04±2.52e+03	6.99e+07±3.46e+07	1.31e+03±2.60e+00
$F_{CEC2017}14$	**1.40e+03±7.24e-01**	1.43e+03±5.01e+00	2.29e+03±6.50e+02	6.17e+03±1.84e+03	2.46e+06±5.24e+06	1.41e+03±3.50e+00
$F_{CEC2017}15$	**1.50e+03±6.21e-01**	1.53e+03±7.19e+00	1.55e+04±1.62e+04	1.81e+04±6.00e+03	8.22e+05±1.46e+06	1.50e+03±5.89e-01
$F_{CEC2017}16$	**1.60e+03±2.34e-01**	1.62e+03±2.03e+01	1.88e+03±1.53e+02	2.12e+03±8.66e+01	2.48e+03±1.42e+02	1.60e+03±1.85e+00
$F_{CEC2017}17$	**1.70e+03±1.44e+00**	1.74e+03±5.98e+00	1.77e+03±4.95e+01	1.82e+03±9.41e+01	2.08e+03±1.37e+02	1.72e+03±6.29e+00
$F_{CEC2017}18$	**1.80e+03±1.83e-01**	1.92e+03±5.70e+01	4.67e+05±7.40e+05	8.44e+03±3.83e+03	7.45e+07±3.45e+07	1.81e+03±2.20e+00
$F_{CEC2017}19$	**1.90e+03±2.52e-01**	1.91e+03±3.72e+00	4.22e+04±5.31e+04	1.61e+04±3.95e+03	1.35e+07±2.23e+07	1.90e+03±4.97e-01
$F_{CEC2017}20$	**2.00e+03±2.63e-01**	2.05e+03±7.31e+00	2.02e+03±8.66e+00	2.27e+03±8.48e+01	2.41e+03±1.06e+02	2.02e+03±7.24e+00
$F_{CEC2017}21$	2.29e+03±5.35e+01	2.20e+03±2.45e-01	2.22e+03±9.15e+00	2.35e+03±1.25e+01	2.39e+03±3.75e+01	**2.19e+03±2.91e+01**
$F_{CEC2017}22$	2.28e+03±3.85e+01	**2.23e+03±2.61e+01**	2.29e+03±3.04e+01	2.30e+03±4.59e-11	3.41e+03±2.50e+02	2.26e+03±2.75e+01
$F_{CEC2017}23$	2.61e+03±5.21e+00	**2.59e+03±8.12e+01**	2.63e+03±9.61e+00	2.71e+03±4.36e+01	2.82e+03±4.65e+01	2.62e+03±4.68e+00
$F_{CEC2017}24$	2.72e+03±6.15e+01	2.51e+03±1.43e+01	2.57e+03±2.96e+01	2.61e+03±1.44e+02	2.96e+03±7.02e+01	**2.50e+03±1.65e+01**
$F_{CEC2017}25$	2.90e+03±1.39e+01	2.90e+03±1.14e+01	2.94e+03±5.30e+01	2.94e+03±1.66e+01	3.46e+03±9.90e+01	**2.72e+03±1.28e+02**
$F_{CEC2017}26$	2.90e+03±0.00e+00	2.79e+03±1.06e+02	3.02e+03±7.48e+01	3.09e+03±4.63e+02	4.41e+03±3.07e+02	**2.71e+03±1.02e+02**
$F_{CEC2017}27$	**3.08e+03±3.40e-01**	3.10e+03±1.75e+00	3.12e+03±2.95e+01	3.20e+03±2.53e+01	3.35e+03±7.37e+01	3.09e+03±1.61e+00
$F_{CEC2017}28$	3.32e+03±4.10e+00	3.07e+03±6.77e+01	3.24e+03±7.12e+01	3.43e+03±6.59e+01	3.92e+03±1.22e+02	**3.00e+03±1.39e+02**
$F_{CEC2017}29$	**3.13e+03±6.06e+00**	3.18e+03±1.14e+01	3.26e+03±6.03e+01	3.34e+03±1.11e+02	3.74e+03±1.37e+02	3.17e+03±1.27e+01
$F_{CEC2017}30$	**3.89e+03±2.58e+02**	5.59e+03±1.81e+03	1.27e+06±8.37e+05	4.09e+05±1.52e+05	5.79e+07±2.83e+07	4.80e+03±9.80e+02
$w/t/l$	8/2/20	23/4/3	28/1/1	26/2/2	30/0/0	-

Table 5.10: Comparison of SFCS with the five metaheuristic algorithms on 30-dimensional benchmark functions from CEC'2013

Function	DE Average±Std	FPA Average±Std	GA Average±Std	GSA Average±Std	SMS Average±Std	SFCS Average±Std
$F_{CEC2013}1$	-1.40e+03±3.71e-04	-1.04e+03±1.96e+02	1.01e+03±6.93e+02	**-1.40e+03±1.89e-13**	5.29e+04±4.37e+03	-1.40e+03±6.78e-03
$F_{CEC2013}2$	2.19e+07±1.67e+07	**2.22e+06±1.06e+06**	1.45e+08±5.30e+07	1.20e+07±2.58e+06	1.00e+08±0.00e+00	9.99e+06±2.24e+06
$F_{CEC2013}3$	3.40e+09±2.77e+09	8.32e+09±2.43e+09	6.09e+10±2.19e+10	8.84e+09±2.48e+09	**1.00e+08±0.00e+00**	1.88e+09±9.49e+08
$F_{CEC2013}4$	1.01e+05±1.16e+04	**2.53e+04±6.65e+03**	1.24e+05±2.07e+04	6.93e+04±4.25e+03	6.42e+04±1.28e+03	7.19e+04±9.96e+03
$F_{CEC2013}5$	**-1.00e+03±3.88e-03**	-8.46e+02±4.63e+01	-6.75e+02±1.38e+02	-9.81e+02±1.42e+01	4.18e+04±1.05e+04	-1.00e+03±1.34e-01
$F_{CEC2013}6$	-8.68e+02±1.85e+01	-7.78e+02±3.24e+01	-5.34e+02±1.02e+02	-8.12e+02±2.27e+01	1.14e+04±2.10e+03	**-8.70e+02±1.17e+01**
$F_{CEC2013}7$	**-6.95e+02±3.11e+01**	-6.81e+02±1.56e+01	-5.47e+02±7.42e+01	-6.92e+02±8.49e+01	1.14e+06±9.46e+05	-6.92e+02±1.51e+01
$F_{CEC2013}8$	-6.79e+02±3.90e-02	**-6.79e+02±6.82e-02**	-6.79e+02±5.69e-02	-6.79e+02±5.00e-02	-6.79e+02±5.30e-02	-6.79e+02±5.01e-02
$F_{CEC2013}9$	-5.60e+02±1.32e+00	-5.66e+02±1.14e+00	-5.65e+02±2.60e+00	-5.66e+02±3.71e+00	-5.54e+02±1.21e+00	**-5.68e+02±2.09e+00**
$F_{CEC2013}10$	**-4.99e+02±2.14e-01**	-4.68e+02±1.16e+01	3.05e+02±2.28e+02	-4.98e+02±8.58e+00	8.07e+03±1.43e+03	-4.98e+02±2.60e-01
$F_{CEC2013}11$	-2.22e+02±2.35e+01	-1.81e+02±2.20e+01	-2.48e+02±2.29e+01	-5.31e+01±2.91e+01	5.00e+02±5.10e+01	**-2.77e+02±1.84e+01**
$F_{CEC2013}12$	-8.23e+01±1.49e+01	-5.49e+01±2.36e+01	1.15e+02±5.68e+01	1.11e+02±4.53e+01	5.62e+02±6.61e+01	**-1.11e+02±2.36e+01**
$F_{CEC2013}13$	**1.95e+01±1.45e+01**	8.82e+01±3.11e+01	2.17e+02±6.67e+01	3.63e+02±5.43e+01	6.65e+02±5.17e+01	2.81e+01±3.18e+01
$F_{CEC2013}14$	6.97e+03±2.47e+02	4.43e+03±2.54e+02	**1.59e+03±3.05e+02**	3.84e+03±6.01e+02	9.30e+03±5.91e+02	2.89e+03±3.61e+02
$F_{CEC2013}15$	7.93e+03±2.87e+02	5.45e+03±2.21e+02	7.29e+03±3.41e+02	**4.00e+03±3.59e+02**	7.53e+03±1.17e+03	4.40e+03±2.92e+02
$F_{CEC2013}16$	2.03e+02±4.01e-01	2.03e+02±2.69e-01	2.03e+02±3.25e-01	**2.00e+02±5.24e-03**	2.02e+02±7.92e-01	2.02e+02±2.72e-01
$F_{CEC2013}17$	5.21e+02±2.31e+01	5.51e+02±2.68e+01	5.94e+02±4.10e+01	**4.12e+02±2.33e+01**	1.12e+03±3.05e+01	4.70e+02±2.22e+01
$F_{CEC2013}18$	6.50e+02±1.34e+01	6.48e+02±2.43e+01	9.32e+02±5.91e+01	**5.07e+02±1.64e+01**	1.21e+03±2.97e+01	6.16e+02±2.27e+01
$F_{CEC2013}19$	5.18e+02±1.18e+00	5.34e+02±1.46e+01	6.49e+02±1.23e+02	5.17e+02±4.76e+00	6.71e+05±1.18e+05	**5.12e+02±2.41e+00**
$F_{CEC2013}20$	6.14e+02±2.82e-01	6.14e+02±3.12e-01	**6.14e+02±4.85e-01**	6.15e+02±7.09e-02	6.15e+02±1.03e-09	6.14e+02±4.44e-01
$F_{CEC2013}21$	1.07e+03±8.82e+01	1.23e+03±3.88e+01	1.82e+03±3.24e+02	1.05e+03±7.04e+01	3.21e+03±2.82e+01	**9.38e+02±2.99e+01**
$F_{CEC2013}22$	8.38e+03±3.16e+02	6.11e+03±2.50e+02	**2.87e+03±3.63e+02**	7.15e+03±6.06e+02	1.08e+04±4.30e+02	4.46e+03±3.88e+02
$F_{CEC2013}23$	9.19e+03±3.11e+02	7.02e+03±2.50e+02	8.45e+03±4.69e+02	6.77e+03±4.77e+02	1.02e+04±3.95e+02	**6.06e+03±3.95e+02**
$F_{CEC2013}24$	1.32e+03±3.94e+00	1.29e+03±8.21e+00	1.32e+03±9.55e+00	1.38e+03±7.12e+01	1.52e+03±8.24e+01	**1.28e+03±7.16e+00**
$F_{CEC2013}25$	**1.42e+03±3.90e+00**	1.45e+03±6.03e+00	1.43e+03±9.83e+00	1.51e+03±1.04e+01	1.55e+03±2.47e+01	1.43e+03±5.66e+00
$F_{CEC2013}26$	1.40e+03±2.50e+00	**1.40e+03±1.22e-01**	1.52e+03±9.24e+01	1.57e+03±1.95e+01	1.62e+03±3.65e+01	1.40e+03±1.94e-01
$F_{CEC2013}27$	2.70e+03±2.59e+01	2.55e+03±5.59e+01	2.67e+03±9.14e+01	**2.32e+03±1.02e+02**	3.13e+03±1.23e+02	2.49e+03±5.09e+01
$F_{CEC2013}28$	**1.70e+03±1.07e+00**	2.89e+03±2.62e+02	3.69e+03±5.35e+02	5.22e+03±3.35e+02	7.91e+03±3.54e+02	1.88e+03±8.97e+01
$w/t/l$	18/5/5	23/0/5	23/2/3	18/2/8	24/1/3	-

Table 5.11: Comparison of SFCS with the five metaheuristic algorithms on 30-dimensional benchmark functions from CEC'2017

Function	DE	FPA	GA	GSA	SMS	SFCS
	Average±Std	Average±Std	Average±Std	Average±Std	Average±Std	Average±Std
$F_{CEC2017}1$	2.65e+04±1.45e+04	2.27e+08±1.16e+08	3.11e+09±9.27e+08	**1.81e+03±1.06e+03**	1.00e+08±0.00e+00	3.68e+05±2.14e+05
$F_{CEC2017}2$	1.63e+25±8.74e+25	8.85e+20±3.55e+21	3.30e+31±9.80e+31	6.01e+25±2.07e+26	**1.00e+08±0.00e+00**	4.87e+16±7.12e+16
$F_{CEC2017}3$	1.28e+05±2.22e+04	**3.18e+04±7.21e+03**	1.99e+05±5.87e+04	9.13e+04±6.88e+03	8.51e+04±2.86e+03	8.03e+04±1.36e+04
$F_{CEC2017}4$	4.97e+02±1.57e+01	5.73e+02±3.52e+01	8.80e+02±1.30e+02	5.44e+02±2.78e+01	1.89e+04±2.95e+03	**4.72e+02±1.86e+01**
$F_{CEC2017}5$	7.18e+02±1.48e+01	6.63e+02±1.31e+01	7.06e+02±3.29e+01	7.32e+02±2.30e+01	9.65e+02±2.23e+01	**6.41e+02±2.89e+01**
$F_{CEC2017}6$	**6.01e+02±3.54e-01**	6.50e+02±5.62e+00	6.14e+02±2.45e+00	6.52e+02±4.98e+00	7.01e+02±1.05e+01	6.44e+02±7.13e+00
$F_{CEC2017}7$	9.49e+02±1.38e+01	9.16e+02±2.18e+01	1.08e+03±3.57e+01	**8.34e+02±1.74e+01**	1.34e+03±6.14e+01	8.65e+02±2.99e+01
$F_{CEC2017}8$	1.02e+03±1.36e+01	9.43e+02±1.40e+01	9.87e+02±2.13e+01	9.56e+02±1.88e+01	1.17e+03±2.72e+01	**9.32e+02±2.16e+01**
$F_{CEC2017}9$	**9.02e+02±1.47e+00**	3.53e+03±6.38e+02	5.17e+03±1.67e+03	3.31e+03±3.66e+02	9.22e+03±3.40e+03	4.59e+03±9.54e+02
$F_{CEC2017}10$	8.58e+03±2.39e+02	5.58e+03±2.29e+02	6.14e+03±5.52e+02	4.87e+03±5.44e+02	7.97e+03±1.03e+03	**4.57e+03±3.27e+02**
$F_{CEC2017}11$	**1.20e+03±1.73e+01**	1.29e+03±3.88e+01	8.89e+03±4.22e+03	1.80e+03±2.76e+02	1.23e+04±6.05e+03	1.21e+03±2.11e+01
$F_{CEC2017}12$	**3.09e+05±2.43e+05**	2.06e+06±1.17e+06	2.23e+08±6.96e+07	2.00e+07±4.28e+07	1.00e+08±0.00e+00	3.25e+05±1.18e+05
$F_{CEC2017}13$	**1.57e+03±5.40e+01**	3.92e+04±1.04e+04	4.23e+07±3.11e+07	3.19e+04±9.09e+03	1.00e+08±0.00e+00	1.24e+04±4.85e+03
$F_{CEC2017}14$	**1.48e+03±1.19e+01**	1.82e+03±1.39e+02	1.85e+06±1.89e+06	7.04e+05±2.75e+05	1.65e+07±8.36e+06	1.64e+03±4.32e+01
$F_{CEC2017}15$	**1.57e+03±8.12e+00**	9.76e+03±3.93e+03	1.90e+07±2.00e+07	1.41e+04±3.31e+03	1.00e+08±0.00e+00	2.49e+03±3.24e+02
$F_{CEC2017}16$	3.35e+03±1.61e+02	3.06e+03±1.29e+02	3.42e+03±3.43e+02	3.24e+03±3.71e+02	7.40e+03±1.17e+03	**2.69e+03±1.39e+02**
$F_{CEC2017}17$	2.13e+03±2.18e+02	2.15e+03±8.75e+01	2.54e+03±2.52e+02	2.84e+03±1.66e+02	9.23e+03±9.02e+03	**1.98e+03±8.44e+01**
$F_{CEC2017}18$	**1.88e+03±1.31e+01**	3.94e+04±1.36e+04	1.11e+07±1.03e+07	3.91e+05±2.20e+05	9.06e+07±1.66e+07	7.11e+04±2.07e+04
$F_{CEC2017}19$	**1.94e+03±6.67e+00**	6.85e+03±2.77e+03	1.69e+07±1.55e+07	2.27e+04±1.19e+04	1.00e+08±0.00e+00	2.11e+03±6.76e+01
$F_{CEC2017}20$	2.61e+03±2.89e+02	2.46e+03±5.74e+01	2.55e+03±1.41e+02	3.07e+03±2.19e+02	3.21e+03±2.26e+02	**2.35e+03±9.78e+01**
$F_{CEC2017}21$	2.51e+03±1.69e+01	2.46e+03±1.33e+01	2.51e+03±2.78e+01	2.59e+03±2.82e+01	2.81e+03±4.98e+01	**2.43e+03±2.44e+01**
$F_{CEC2017}22$	9.32e+03±1.24e+03	2.81e+03±1.96e+02	4.19e+03±2.27e+03	6.19e+03±1.89e+03	9.99e+03±5.54e+02	**2.37e+03±4.81e+01**
$F_{CEC2017}23$	2.88e+03±1.62e+01	3.01e+03±4.06e+01	**2.87e+03±2.67e+01**	3.65e+03±1.81e+02	3.86e+03±2.25e+02	2.87e+03±3.44e+01
$F_{CEC2017}24$	3.06e+03±1.46e+01	3.15e+03±4.41e+01	3.12e+03±4.83e+01	3.32e+03±8.70e+01	4.20e+03±2.52e+02	**2.99e+03±3.54e+01**
$F_{CEC2017}25$	**2.88e+03±1.03e+00**	2.93e+03±1.60e+01	3.20e+03±1.15e+02	2.94e+03±1.32e+01	5.88e+03±4.43e+02	2.89e+03±1.03e+00
$F_{CEC2017}26$	5.73e+03±1.88e+02	5.36e+03±8.44e+02	6.15e+03±6.31e+02	6.76e+03±1.49e+03	1.17e+04±5.53e+02	**3.38e+03±3.76e+02**
$F_{CEC2017}27$	**3.30e+03±2.06e+01**	3.43e+03±2.66e+01	3.31e+03±2.97e+01	4.79e+03±3.85e+02	5.14e+03±3.67e+02	3.31e+03±1.60e+01
$F_{CEC2017}28$	3.22e+03±1.11e+01	3.29e+03±2.49e+01	3.56e+03±1.04e+02	3.35e+03±6.18e+01	7.66e+03±4.84e+02	**3.22e+03±8.09e+00**
$F_{CEC2017}29$	4.45e+03±2.06e+02	4.28e+03±1.28e+02	4.30e+03±2.78e+02	4.80e+03±2.26e+02	8.85e+03±1.50e+03	**4.00e+03±1.38e+02**
$F_{CEC2017}30$	**3.51e+04±1.47e+04**	1.22e+05±5.65e+04	2.56e+07±2.51e+07	5.86e+05±4.86e+05	1.00e+08±0.00e+00	6.43e+04±2.86e+04
$w/t/l$	16/2/12	27/0/3	26/3/1	27/0/3	29/0/1	-

Table 5.12: Comparison of SFCS with the five metaheuristic algorithms on 50-dimensional benchmark functions from CEC'2013

Function	DE Average±Std	FPA Average±Std	GA Average±Std	GSA Average±Std	SMS Average±Std	SFCS Average±Std
$F_{CEC2013}1$	-1.40e+03±1.10e+00	3.25e+03±1.17e+03	1.35e+04±2.47e+03	**-1.40e+03±0.00e+00**	7.68e+04±2.37e+03	-1.39e+03±2.67e+00
$F_{CEC2013}2$	2.39e+08±1.12e+08	**1.82e+07±6.35e+06**	4.68e+08±1.39e+08	2.25e+07±8.94e+06	1.00e+08±0.00e+00	2.66e+07±5.65e+06
$F_{CEC2013}3$	5.88e+10±2.39e+10	3.58e+10±8.04e+09	1.75e+13±9.49e+13	2.87e+10±7.79e+09	**1.00e+08±0.00e+00**	1.40e+10±3.74e+09
$F_{CEC2013}4$	1.98e+05±1.76e+04	**5.86e+04±7.13e+03**	2.04e+05±3.77e+04	9.15e+04±6.18e+03	9.60e+04±1.00e+04	1.26e+05±1.30e+04
$F_{CEC2013}5$	**-9.97e+02±1.26e+00**	-1.59e+02±2.31e+02	3.15e+02±4.81e+02	-8.08e+02±4.15e+01	1.90e+04±2.85e+03	-9.84e+02±4.40e+00
$F_{CEC2013}6$	**-8.52e+02±1.17e+00**	-5.58e+02±3.86e+01	3.62e+02±3.77e+02	-7.17e+02±4.46e+01	9.32e+03±1.03e+03	-8.47e+02±4.81e+00
$F_{CEC2013}7$	-6.29e+02±3.29e+01	-6.61e+02±1.41e+01	-5.21e+02±5.67e+01	**-6.86e+02±2.39e+01**	3.89e+04±2.73e+04	-6.71e+02±1.26e+01
$F_{CEC2013}8$	**-6.79e+02±4.91e-02**	-6.79e+02±3.77e-02	-6.79e+02±3.93e-02	-6.79e+02±4.37e-02	-6.79e+02±3.77e-02	-6.79e+02±2.90e-02
$F_{CEC2013}9$	-5.24e+02±1.16e+00	-5.35e+02±1.79e+00	-5.30e+02±2.78e+00	**-5.47e+02±3.17e+00**	-5.21e+02±1.83e+00	-5.39e+02±2.19e+00
$F_{CEC2013}10$	-4.15e+02±3.98e+01	2.98e+00±1.38e+02	2.65e+03±7.15e+02	-3.85e+02±6.61e+01	1.29e+04±9.93e+02	**-4.60e+02±1.02e+01**
$F_{CEC2013}11$	-1.22e+01±2.51e+01	6.84e+01±2.72e+01	7.85e+01±4.63e+01	1.35e+02±5.52e+01	7.53e+02±7.80e+01	**-1.09e+02±3.64e+01**
$F_{CEC2013}12$	**1.40e+02±1.96e+01**	2.34e+02±4.11e+01	5.86e+02±1.23e+02	5.16e+02±6.69e+01	9.32e+02±5.15e+01	1.42e+02±4.65e+01
$F_{CEC2013}13$	**2.49e+02±2.33e+01**	4.05e+02±4.90e+01	6.60e+02±9.33e+01	7.85e+02±8.34e+01	1.04e+03±3.19e+01	3.24e+02±5.55e+01
$F_{CEC2013}14$	1.33e+04±3.37e+02	9.82e+03±4.00e+02	**5.32e+03±4.51e+02**	6.96e+03±6.93e+02	1.65e+04±7.66e+02	7.19e+03±5.13e+02
$F_{CEC2013}15$	1.49e+04±2.82e+02	1.17e+04±3.72e+02	1.47e+04±5.32e+02	**8.85e+03±6.68e+02**	1.57e+04±5.91e+02	9.93e+03±3.88e+02
$F_{CEC2013}16$	2.04e+02±3.26e-01	2.04e+02±3.69e-01	2.04e+02±3.88e-01	**2.00e+02±5.42e-03**	2.03e+02±9.13e-01	2.04e+02±2.73e-01
$F_{CEC2013}17$	7.51e+02±2.83e+01	8.77e+02±3.92e+01	1.25e+03±7.54e+01	**6.92e+02±5.14e+01**	1.43e+03±5.28e+01	7.21e+02±3.05e+01
$F_{CEC2013}18$	8.85e+02±2.30e+01	9.78e+02±4.45e+01	1.70e+03±1.51e+02	**7.87e+02±4.66e+01**	1.59e+03±5.54e+01	9.15e+02±3.53e+01
$F_{CEC2013}19$	5.41e+02±3.78e+00	1.43e+03±4.81e+02	6.75e+03±3.10e+03	6.89e+02±5.39e+01	1.45e+06±3.26e+05	**5.41e+02±6.27e+00**
$F_{CEC2013}20$	6.25e+02±1.62e-01	**6.24e+02±2.84e-01**	6.24e+02±3.60e-01	6.25e+02±2.98e-01	6.25e+02±1.23e-02	6.24e+02±3.48e-01
$F_{CEC2013}21$	**1.15e+03±3.77e+02**	2.54e+03±2.19e+02	4.05e+03±5.05e+02	2.20e+03±3.65e+02	5.13e+03±3.54e+01	1.16e+03±9.95e+01
$F_{CEC2013}22$	1.50e+04±4.41e+02	1.23e+04±3.84e+02	**6.84e+03±6.74e+02**	1.33e+04±5.01e+02	1.86e+04±4.06e+02	9.81e+03±5.76e+02
$F_{CEC2013}23$	1.62e+04±3.87e+02	1.38e+04±3.71e+02	1.62e+04±5.18e+02	1.28e+04±6.97e+02	1.79e+04±6.48e+02	**1.24e+04±5.16e+02**
$F_{CEC2013}24$	1.41e+03±5.32e+00	1.40e+03±1.03e+01	1.41e+03±1.22e+01	1.62e+03±1.17e+02	2.00e+03±1.69e+02	**1.36e+03±1.28e+01**
$F_{CEC2013}25$	**1.52e+03±4.89e+00**	1.61e+03±8.60e+00	1.54e+03±1.33e+01	1.74e+03±2.11e+01	1.78e+03±3.46e+01	1.57e+03±1.33e+01
$F_{CEC2013}26$	1.69e+03±9.35e+00	1.43e+03±2.93e+01	1.68e+03±4.29e+01	1.64e+03±6.32e+01	1.76e+03±4.02e+01	**1.41e+03±4.54e+00**
$F_{CEC2013}27$	3.62e+03±4.41e+01	3.56e+03±5.61e+01	3.58e+03±1.00e+02	3.40e+03±1.36e+02	4.58e+03±3.31e+02	**3.40e+03±8.93e+01**
$F_{CEC2013}28$	**1.82e+03±8.00e+00**	3.72e+03±4.58e+02	5.25e+03±1.37e+03	9.51e+03±4.10e+02	1.25e+04±5.97e+02	1.89e+03±2.03e+01
$w/t/l$	17/3/8	23/3/2	24/1/3	16/3/9	24/1/3	-

Table 5.13: Comparison of SFCS with the five metaheuristic algorithms on 50-dimensional benchmark functions from CEC'2017

Function	DE Average±Std	FPA Average±Std	GA Average±Std	GSA Average±Std	SMS Average±Std	SFCS Average±Std
$F_{CEC2017}1$	5.35e+06±3.36e+06	7.03e+09±1.60e+09	2.13e+10±4.22e+09	**1.37e+06±7.08e+06**	1.00e+08±0.00e+00	2.61e+07±1.01e+07
$F_{CEC2017}2$	1.59e+64±8.70e+64	1.11e+50±5.94e+50	1.66e+63±7.04e+63	7.13e+59±2.02e+60	**1.00e+08±0.00e+00**	1.07e+43±4.15e+43
$F_{CEC2017}3$	3.55e+05±3.64e+04	**1.15e+05±2.44e+04**	4.01e+05±8.90e+04	1.84e+05±1.43e+04	2.08e+05±2.50e+04	2.27e+05±2.20e+04
$F_{CEC2017}4$	6.19e+02±3.35e+01	1.45e+03±2.05e+02	3.41e+03±6.67e+02	7.47e+02±1.74e+02	3.81e+04±2.90e+03	**5.90e+02±3.26e+01**
$F_{CEC2017}5$	9.40e+02±2.07e+01	8.63e+02±2.05e+01	9.85e+02±3.15e+01	8.25e+02±1.87e+01	1.18e+03±4.96e+01	**8.03e+02±4.04e+01**
$F_{CEC2017}6$	**6.07e+02±1.90e+00**	6.65e+02±4.81e+00	6.31e+02±3.76e+00	6.61e+02±2.41e+00	7.11e+02±8.51e+00	6.60e+02±6.32e+00
$F_{CEC2017}7$	1.19e+03±1.62e+01	1.23e+03±4.91e+01	1.73e+03±1.13e+02	1.17e+03±4.50e+01	1.74e+03±5.58e+01	**1.12e+03±4.38e+01**
$F_{CEC2017}8$	1.25e+03±2.00e+01	1.17e+03±2.76e+01	1.27e+03±5.10e+01	1.16e+03±2.33e+01	1.51e+03±5.44e+01	**1.12e+03±3.32e+01**
$F_{CEC2017}9$	**1.38e+03±2.83e+02**	1.52e+04±2.69e+03	2.29e+04±5.35e+03	1.09e+04±1.08e+03	2.41e+04±1.04e+04	1.74e+04±3.11e+03
$F_{CEC2017}10$	1.52e+04±3.80e+02	1.04e+04±3.02e+02	1.21e+04±5.54e+02	**8.13e+03±7.62e+02**	1.61e+04±6.61e+02	8.42e+03±4.02e+02
$F_{CEC2017}11$	**1.36e+03±2.83e+01**	2.03e+03±1.73e+02	1.85e+04±7.86e+03	7.03e+03±1.65e+03	2.47e+04±2.87e+03	1.55e+03±8.00e+01
$F_{CEC2017}12$	1.36e+07±6.51e+06	4.54e+08±3.09e+08	2.92e+09±9.98e+08	1.18e+07±3.06e+07	1.00e+08±0.00e+00	**9.53e+06±2.78e+06**
$F_{CEC2017}13$	**2.19e+04±6.78e+03**	6.71e+06±4.58e+06	3.79e+08±1.95e+08	3.56e+04±9.07e+03	1.00e+08±0.00e+00	7.51e+04±3.58e+04
$F_{CEC2017}14$	**1.60e+03±1.60e+01**	1.69e+04±1.30e+04	1.10e+07±1.23e+07	1.81e+06±8.06e+05	9.68e+07±9.51e+06	3.78e+04±2.20e+04
$F_{CEC2017}15$	**2.00e+03±9.60e+01**	6.48e+04±3.44e+04	1.06e+08±6.35e+07	1.59e+06±8.59e+06	1.00e+08±0.00e+00	1.02e+04±2.83e+03
$F_{CEC2017}16$	5.40e+03±2.38e+02	4.24e+03±1.80e+02	5.21e+03±3.66e+02	3.85e+03±4.14e+02	1.10e+04±1.62e+03	**3.55e+03±2.07e+02**
$F_{CEC2017}17$	4.15e+03±2.17e+02	3.30e+03±1.39e+02	4.25e+03±3.40e+02	3.65e+03±4.00e+02	2.11e+04±7.76e+03	**3.00e+03±1.75e+02**
$F_{CEC2017}18$	**6.39e+04±5.42e+04**	2.19e+05±1.18e+05	3.46e+07±1.88e+07	2.83e+06±1.39e+06	9.83e+07±7.67e+06	1.09e+06±4.87e+05
$F_{CEC2017}19$	**2.03e+03±1.53e+01**	9.32e+04±4.47e+04	3.10e+07±1.82e+07	1.03e+05±3.44e+04	1.00e+08±0.00e+00	1.51e+04±4.38e+03
$F_{CEC2017}20$	4.24e+03±1.88e+02	3.35e+03±9.99e+01	3.50e+03±2.30e+02	3.56e+03±3.70e+02	4.30e+03±5.30e+02	**2.99e+03±1.63e+02**
$F_{CEC2017}21$	2.74e+03±1.87e+01	2.69e+03±2.82e+01	2.79e+03±4.78e+01	2.80e+03±3.96e+01	3.30e+03±5.99e+01	**2.62e+03±4.06e+01**
$F_{CEC2017}22$	1.64e+04±3.34e+02	1.22e+04±2.57e+02	1.37e+04±7.52e+02	1.11e+04±6.48e+02	1.80e+04±4.81e+02	**1.01e+04±1.22e+03**
$F_{CEC2017}23$	**3.27e+03±3.28e+01**	3.62e+03±6.60e+01	3.29e+03±4.67e+01	4.56e+03±1.73e+02	5.05e+03±3.04e+02	3.29e+03±4.65e+01
$F_{CEC2017}24$	3.49e+03±5.18e+01	3.77e+03±5.74e+01	3.57e+03±6.37e+01	3.88e+03±6.66e+01	5.67e+03±2.92e+02	**3.35e+03±6.62e+01**
$F_{CEC2017}25$	3.13e+03±4.47e+01	3.60e+03±1.59e+02	5.41e+03±7.49e+02	3.33e+03±1.60e+02	1.57e+04±7.22e+02	**3.08e+03±2.54e+01**
$F_{CEC2017}26$	8.40e+03±2.24e+02	1.03e+04±4.51e+02	9.63e+03±5.73e+02	1.12e+04±1.27e+03	1.73e+04±5.14e+02	**8.17e+03±8.43e+02**
$F_{CEC2017}27$	4.16e+03±1.21e+02	4.56e+03±1.48e+02	**3.93e+03±1.13e+02**	7.48e+03±6.27e+02	8.27e+03±8.05e+02	4.02e+03±1.26e+02
$F_{CEC2017}28$	3.49e+03±8.30e+01	4.14e+03±2.31e+02	5.81e+03±8.74e+02	3.92e+03±1.57e+02	1.42e+04±7.00e+02	**3.41e+03±3.68e+01**
$F_{CEC2017}29$	5.96e+03±2.90e+02	5.93e+03±3.27e+02	5.66e+03±5.05e+02	6.00e+03±4.71e+02	2.27e+05±2.18e+05	**4.93e+03±2.25e+02**
$F_{CEC2017}30$	**6.81e+06±1.74e+06**	3.96e+07±9.21e+06	1.12e+08±8.82e+07	9.86e+07±1.54e+07	1.00e+08±0.00e+00	1.23e+07±1.95e+06
$w/t/l$	18/2/10	26/0/4	27/1/2	23/1/6	28/0/2	-

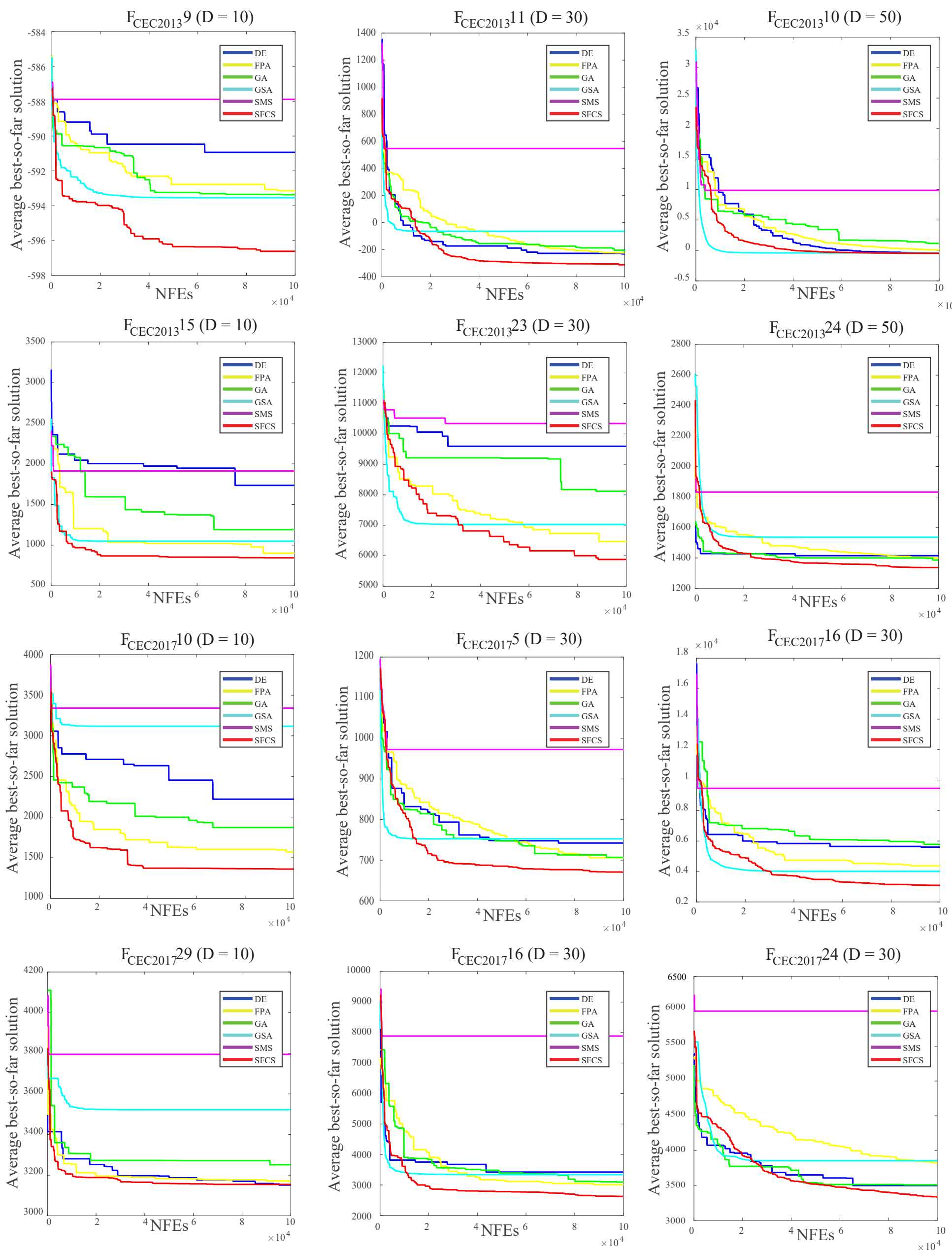

Figure 5.2: Convergence graphs of SFCS and the five metaheuristic algorithms on 12 randomly-selected benchmark functions.

Table 5.14: Statistical analysis of the SFCSs with different values of M_0 by the Friedman's test

M_0	Ranking	$z-value$	Unadjusted p	p_{Bonf}
4	3.2931	0.00000	1.00000	5.00000
5	3.2931	-	-	-
6	3.5086	0.62037	0.53502	2.67509
7	3.6207	0.94295	0.34570	1.72852
8	3.8966	1.73702	0.08238	0.41192
9	3.3879	0.27296	0.78488	3.92442

precisely affects the topological structure. Hence, the performances of the SFCS algorithm on CEC'2017 special session with different values of M_0 are examined, and the corresponding results are presented in Table 5.16 and Table 5.17. M_0 is assigned to be 4, 5, 6, 7, 8 and 9. It is obvious that the SFCSs with these values of M_0 achieve the best solutions on 12, 13, 5, 10, 7 and 11 functions accordingly. When M_0 is 5, the SFCS obtains the best solution on the maximum number of functions. The performances of SFCS with $M_0 = 4$ rank in the second, and $M_0 = 6$ presents the worst performance. Moreover, Friedman's test is utilized to distinguish the significance difference between each pair of SFCS architecture design, and the results are provided in Table 5.14 [107]. We can observe that the SFCS with $M_0 = 5$ and $M_0 = 4$ achieves more satisfactory performance than the SFCS with other values of M_0. The performance of the SFCS with $M_0 = 9$ ranks second. $M_0 = 6$ has the third-best result, while $M_0 = 8$ has the worst. In addition, the Bonferroni-Dunn procedure is utilized as the post hoc test to describe differences obtained by statistical tests [106]. The results illustrate that all adjusted $p-values$ are larger than 0.05. In other words, there is no statistical significance difference when M_0 is assigned different values. Thus, it can be concluded that the value setting of M_0 has little influence on the superiority of the SFCS algorithm.

5.6.2 Real-world optimization tasks

We adopt 21 real-world optimization tasks from CEC'2011, and more properties are summarized in Table 5.15. [134].The performance of the SFCS adopted in optimizing real-world problems is compared with the basic CS algorithm and two CS variants. From Table 5.18, we find that the CS, PSOCS, MCS and SFCS obtain the best

Table 5.15: Description of the real-world benchmark problems from CEC'2011

Function	Dimension	Constraints
$F_{CEC2011}1$	6	Bound constrained
$F_{CEC2011}2$	30	Bound constrained
$F_{CEC2011}3$	1	Bound constrained
$F_{CEC2011}4$	1	Unconstrained
$F_{CEC2011}5$	30	Bound constrained
$F_{CEC2011}6$	30	Bound constrained
$F_{CEC2011}7$	20	Bound constrained
$F_{CEC2011}8$	7	Equality and inequality constraints
$F_{CEC2011}9$	126	Linear equality constraints
$F_{CEC2011}10$	12	Bound constrained
$F_{CEC2011}11$	120	Inequality constraints
$F_{CEC2011}12$	216	Inequality constraints
$F_{CEC2011}13$	6	Inequality constraints
$F_{CEC2011}14$	13	Inequality constraints
$F_{CEC2011}15$	15	Inequality constraints
$F_{CEC2011}16$	40	Inequality constraints
$F_{CEC2011}17$	140	Inequality constraints
$F_{CEC2011}18$	96	Inequality constraints
$F_{CEC2011}19$	96	Inequality constraints
$F_{CEC2011}20$	96	Inequality constraints
$F_{CEC2011}21$	26	Bound constrained

solutions on 2, 0, 7 and 12 problems, respectively. It is easy to observe that the SFCS is superior to the other methods in terms of *Average* and *Std*. It is obvious that the SFCS significantly outperforms CS on 11 problems and performs similarly on 10 problems. For the PSOCS, the SFCS evidently outperforms the CS on all 21 problems. Compared with MCS, the SFCS clearly obtains better performances on 14 problems and worse performances on the remaining 7 problems. The results indicate that SFCS can solve real-world problems better than CS and CS variants.

The comparison results are provided in Table 5.19. The SFCS achieves the best performance on 12 problems, and the DE, FPA, GA, GSA and SMS have the best performance on 5, 2, 0, 1 and 1 problems, respectively. Moreover, the statistical results suggest that the SFCS is significantly superior on 15 problems and similar on 1 problem compared with the DE. In contrast, with FPA, the SFCS has better results on 15 problems and worse results on 1 problem. Compared with the GA, the number of the problems on which the SFCS has distinctly better results is 19, and a worse result on only 1. For the GSA, the SFCS behaves better on 15 problems and similarly on 2 problems. The SFCS obviously outperforms the SMS on 20 problems, and the $F_{CEC2011}2$ problem is the only exception in which SMS has more satisfactory performance than SFCS. Therefore, SFCS is an excellent and effective algorithm for solving real-world tasks, which verifies that the scale-free network architecture design enhances the searching ability of SFCS for engineering applications effectively.

5.6.3 Extension

Additionally the SFIPSO is employed to compare with the scale-free architecture design [90]. From Table 5.20 and Table 5.21, SFCS performs better than the SFIPSO on 53 (out of 58) 10-dimensional functions from CEC'2013 and CEC'2017 in terms of *Average* and *Std*. In the experiments of 30 and 50 dimensional functions, the SFCS obtains better solutions on 54 and 53 (out of 58) functions, respectively. The statistical analysis also shows that SFCS obviously outperforms the SFIPSO on 50, 51 and 51 (out of 58) functions, separately. In addition, the computational complexity of the PSO in the worst case is $O(T*(2N)+2N)$, while that of the SFIPSO is $O(T*(2N*\log{(N)}+2N)+2N)$. The simplified results are $O(T*N)$ and $O(T*N*\log{(N)})$. The computational complexity of SFIPSO is obviously larger than that of the PSO, which means that the SFIPSO costs more computational resources to solve the same problem. Our proposed scale-free population topology method is capable of improving the searching ability of the SFCS without broadening the computational complexity owing to its simplified structure.

Furthermore, the influence of the scale-free population topology on other population-based algorithms is also considered. Therefore, we introduce the scale-free population topology into the DE and FA [135] and propose the DE with scale-free population topology (SFDE) and FA with scale-free population topology (SFFA). In contrast to the original DE and FA, the individuals are updated by using the following search strategy:

$$X_i^{t+1} = X_i^t + rand(0,1)(X_{neighbor}^t - X_i^t). \tag{5.1}$$

Eq. (5.1) is similar to Eq. (3.3). Note that the population have to be ranked according to their fitness in each iteration. As shown in Table 5.22 and Table 5.23, although the SFDE achieves 18 (out of 58) better results than the DE in the experiments on 10-dimensional functions from CEC'2013 and CEC'2017, it outperforms the DE on

43 (out of 58) 30-dimensional functions and 48 (out of 58) 50-dimensional functions. The statistical analysis shows that the SFDE is clearly better than DE on 9, 34 and 45 (out of 58) functions, respectively. In the experiments on low-dimensional benchmark functions, the SFDE can not achieve the ideal performance. As the dimension of the functions increases, the ability of the SFDE can be gradually exerted. The comparison results of FA and SFFA are provided in Table 5.24 and Table 5.25, SFFA obtains better results than the FA on most functions and the only exception can be found that the performance of the SFFA is only slightly inferior to the FA in terms of *Average* while the SFFA achieves a better result than the FA in terms of *Std* on $F_{CEC2013}21$. Moreover, the SFFA significantly outperforms the FA on most cases. The only exception is that the SFFA and FA perform similar on 1 30-dimensional function from CEC'2017. Thus, scale-free population topology architecture design can not only improve the performance of CS but also be beneficial to other algorithms. We can believe that the scale-free population topology might be a promising mechanism in enhancing the search ability of population-based algorithms.

Table 5.16: Comparison of the SFCS with different values of M_0 on the benchmark functions from CEC'2013

Function	$M_0 = 4$ Average±Std	$M_0 = 5$ Average±Std	$M_0 = 6$ Average±Std	$M_0 = 7$ Average±Std	$M_0 = 8$ Average±Std	$M_0 = 9$ Average±Std
$F_{CEC2013}1$	-1.40e+03±7.72e-03	-1.40e+03±8.55e-03	-1.40e+03±8.85e-03	-1.40e+03±8.86e-03	**-1.40e+03±8.24e-03**	-1.40e+03±9.89e-03
$F_{CEC2013}2$	9.82e+06±2.80e+06	1.06e+07±2.88e+06	**9.72e+06±2.67e+06**	1.01e+07±2.90e+06	1.01e+07±2.70e+06	1.05e+07±3.36e+06
$F_{CEC2013}3$	1.94e+09±9.93e+08	**1.65e+09±8.35e+08**	1.78e+09±7.19e+08	1.75e+09±5.80e+08	2.01e+09±7.38e+08	2.21e+09±1.05e+09
$F_{CEC2013}4$	6.95e+04±1.16e+04	6.73e+04±9.24e+03	6.71e+04±1.06e+04	**6.64e+04±9.38e+03**	6.97e+04±1.25e+04	6.93e+04±1.06e+04
$F_{CEC2013}5$	**-1.00e+03±1.11e-01**	-1.00e+03±1.34e-01	-1.00e+03±1.06e-01	-1.00e+03±1.82e-01	-1.00e+03±1.54e-01	-1.00e+03±1.30e-01
$F_{CEC2013}6$	-8.64e+02±1.55e+01	-8.67e+02±1.37e+01	**-8.69e+02±1.76e+01**	-8.66e+02±1.46e+01	-8.68e+02±1.34e+01	-8.66e+02±1.62e+01
$F_{CEC2013}7$	-6.97e+02±1.43e+01	-6.95e+02±9.92e+00	-6.96e+02±1.35e+01	-6.96e+02±1.71e+01	-6.93e+02±1.46e+01	**-6.97e+02±1.33e+01**
$F_{CEC2013}8$	-6.79e+02±4.86e-02	-6.79e+02±4.79e-02	-6.79e+02±4.77e-02	-6.79e+02±4.69e-02	-6.79e+02±3.88e-02	**-6.79e+02±4.58e-02**
$F_{CEC2013}9$	-5.69e+02±1.76e+00	-5.69e+02±1.89e+00	-5.69e+02±1.88e+00	-5.69e+02±2.18e+00	-5.69e+02±1.49e+00	**-5.69e+02±1.92e+00**
$F_{CEC2013}10$	-4.98e+02±2.01e-01	-4.98e+02±2.20e-01	-4.98e+02±3.19e-01	-4.98e+02±3.06e-01	**-4.98e+02±2.67e-01**	-4.98e+02±3.29e-01
$F_{CEC2013}11$	-2.81e+02±2.22e+01	**-2.83e+02±2.04e+01**	-2.79e+02±1.91e+01	-2.69e+02±1.69e+01	-2.82e+02±1.58e+01	-2.81e+02±1.74e+01
$F_{CEC2013}12$	-1.01e+02±2.46e+01	**-1.16e+02±3.16e+01**	-1.09e+02±3.28e+01	-9.66e+01±2.90e+01	-1.14e+02±3.13e+01	-1.08e+02±2.59e+01
$F_{CEC2013}13$	**2.34e+01±3.01e+01**	2.66e+01±2.31e+01	3.25e+01±2.92e+01	2.89e+01±3.27e+01	3.85e+01±2.73e+01	3.49e+01±2.43e+01
$F_{CEC2013}14$	**2.86e+03±2.94e+02**	2.91e+03±2.91e+02	2.89e+03±2.94e+02	3.05e+03±3.67e+02	2.98e+03±3.09e+02	2.95e+03±3.49e+02
$F_{CEC2013}15$	4.46e+03±3.59e+02	4.37e+03±2.70e+02	4.54e+03±3.15e+02	4.46e+03±2.72e+02	4.42e+03±3.66e+02	**4.32e+03±3.25e+02**
$F_{CEC2013}16$	2.03e+02±3.50e-01	**2.02e+02±2.98e-01**	2.02e+02±3.22e-01	2.02e+02±3.23e-01	2.02e+02±2.85e-01	2.02e+02±3.08e-01
$F_{CEC2013}17$	4.68e+02±1.64e+01	4.73e+02±1.82e+01	4.70e+02±1.85e+01	**4.63e+02±1.79e+01**	4.66e+02±1.80e+01	4.72e+02±1.62e+01
$F_{CEC2013}18$	**6.15e+02±2.16e+01**	6.18e+02±2.24e+01	6.18e+02±2.07e+01	6.23e+02±2.50e+01	6.20e+02±2.35e+01	6.19e+02±2.29e+01
$F_{CEC2013}19$	5.12e+02±2.13e+00	5.12e+02±2.15e+00	5.12e+02±1.90e+00	5.12e+02±1.87e+00	5.12e+02±2.19e+00	**5.12e+02±1.85e+00**
$F_{CEC2013}20$	6.14e+02±3.41e-01	**6.14e+02±6.11e-01**	6.14e+02±4.03e-01	6.14e+02±2.44e-01	6.14e+02±3.57e-01	6.14e+02±2.90e-01
$F_{CEC2013}21$	9.48e+02±2.82e+01	9.44e+02±3.01e+01	**9.34e+02±2.83e+01**	9.37e+02±1.47e+01	9.38e+02±2.54e+01	9.36e+02±1.32e+01
$F_{CEC2013}22$	4.46e+03±3.70e+02	4.43e+03±3.87e+02	4.56e+03±4.04e+02	**4.31e+03±3.54e+02**	4.49e+03±3.48e+02	4.53e+03±4.62e+02
$F_{CEC2013}23$	6.11e+03±3.30e+02	5.89e+03±4.36e+02	6.00e+03±3.90e+02	**5.84e+03±4.95e+02**	6.11e+03±3.61e+02	5.93e+03±3.68e+02
$F_{CEC2013}24$	1.28e+03±8.95e+00	**1.28e+03±8.01e+00**	1.28e+03±7.02e+00	1.28e+03±7.56e+00	1.28e+03±6.70e+00	1.28e+03±8.18e+00
$F_{CEC2013}25$	1.43e+03±6.44e+00	**1.43e+03±9.32e+00**	1.43e+03±7.45e+00	1.43e+03±8.60e+00	1.43e+03±7.42e+00	1.43e+03±8.54e+00
$F_{CEC2013}26$	1.40e+03±2.37e-01	1.40e+03±2.19e-01	1.40e+03±2.00e-01	1.40e+03±2.31e-01	**1.40e+03±1.54e-01**	1.40e+03±2.07e-01
$F_{CEC2013}27$	**2.47e+03±5.04e+01**	2.49e+03±5.93e+01	2.50e+03±6.20e+01	2.49e+03±6.62e+01	2.48e+03±6.66e+01	2.48e+03±4.58e+01
$F_{CEC2013}28$	1.88e+03±8.71e+01	1.88e+03±6.02e+01	1.88e+03±5.90e+01	**1.86e+03±7.99e+01**	1.88e+03±6.16e+01	1.88e+03±7.37e+01

Table 5.17: Comparison of the SFCS with different values of M_0 on the benchmark functions from CEC'2017

Function	$M_0 = 4$ Average±Std	$M_0 = 5$ Average±Std	$M_0 = 6$ Average±Std	$M_0 = 7$ Average±Std	$M_0 = 8$ Average±Std	$M_0 = 9$ Average±Std
$F_{CEC2017}1$	3.26e+05±1.15e+05	4.26e+05±1.96e+05	**3.15e+05±1.05e+05**	3.19e+05±9.05e+04	4.11e+05±1.78e+05	3.79e+05±1.87e+05
$F_{CEC2017}2$	9.01e+16±2.04e+17	**2.85e+16±3.08e+16**	3.97e+16±5.44e+16	7.98e+16±1.13e+17	9.42e+16±2.08e+17	1.06e+17±1.88e+17
$F_{CEC2017}3$	7.74e+04±1.18e+04	8.02e+04±1.27e+04	7.88e+04±1.45e+04	8.08e+04±1.24e+04	8.28e+04±1.20e+04	**7.61e+04±1.20e+04**
$F_{CEC2017}4$	4.77e+02±1.22e+01	4.75e+02±1.91e+01	4.77e+02±2.01e+01	4.75e+02±1.55e+01	4.79e+02±1.71e+01	**4.71e+02±1.72e+01**
$F_{CEC2017}5$	6.45e+02±2.01e+01	**6.37e+02±2.43e+01**	6.40e+02±2.43e+01	6.50e+02±2.67e+01	6.51e+02±2.11e+01	6.38e+02±2.40e+01
$F_{CEC2017}6$	**6.43e+02±6.80e+00**	6.45e+02±7.26e+00	6.44e+02±6.53e+00	6.44e+02±6.11e+00	6.44e+02±4.39e+00	6.45e+02±6.00e+00
$F_{CEC2017}7$	8.70e+02±2.43e+01	8.69e+02±2.57e+01	8.68e+02±2.43e+01	**8.63e+02±2.56e+01**	8.68e+02±1.95e+01	8.69e+02±2.39e+01
$F_{CEC2017}8$	9.27e+02±1.64e+01	9.31e+02±2.66e+01	9.29e+02±1.69e+01	9.23e+02±2.08e+01	**9.22e+02±2.31e+01**	9.32e+02±2.32e+01
$F_{CEC2017}9$	**4.50e+03±9.47e+02**	4.50e+03±8.55e+02	4.58e+03±8.61e+02	4.78e+03±1.12e+03	4.69e+03±1.03e+03	4.73e+03±9.12e+02
$F_{CEC2017}10$	4.55e+03±3.23e+02	4.54e+03±3.49e+02	4.52e+03±2.97e+02	**4.46e+03±3.09e+02**	4.64e+03±2.59e+02	4.55e+03±2.90e+02
$F_{CEC2017}11$	1.22e+03±3.09e+01	1.22e+03±2.25e+01	1.22e+03±1.94e+01	**1.22e+03±1.96e+01**	1.22e+03±1.96e+01	1.22e+03±2.43e+01
$F_{CEC2017}12$	3.29e+05±1.60e+05	3.39e+05±1.33e+05	3.71e+05±1.67e+05	**3.26e+05±1.36e+05**	3.57e+05±1.47e+05	3.54e+05±1.38e+05
$F_{CEC2017}13$	1.20e+04±2.54e+03	1.18e+04±2.38e+03	1.19e+04±2.72e+03	1.21e+04±3.23e+03	1.18e+04±3.50e+03	**1.17e+04±3.25e+03**
$F_{CEC2017}14$	**1.62e+03±3.97e+01**	1.63e+03±3.97e+01	1.64e+03±5.04e+01	1.64e+03±4.16e+01	1.62e+03±3.56e+01	1.62e+03±3.01e+01
$F_{CEC2017}15$	**2.52e+03±3.17e+02**	2.55e+03±2.77e+02	2.60e+03±3.46e+02	2.55e+03±3.40e+02	2.54e+03±2.69e+02	2.55e+03±3.89e+02
$F_{CEC2017}16$	2.60e+03±1.41e+02	2.66e+03±1.92e+02	2.60e+03±1.54e+02	2.63e+03±1.45e+02	2.67e+03±1.61e+02	**2.57e+03±1.15e+02**
$F_{CEC2017}17$	2.00e+03±9.34e+01	**1.99e+03±9.93e+01**	2.00e+03±1.10e+02	1.99e+03±1.10e+02	2.01e+03±9.08e+01	2.01e+03±1.18e+02
$F_{CEC2017}18$	7.06e+04±2.38e+04	6.38e+04±2.41e+04	6.48e+04±1.67e+04	7.05e+04±2.55e+04	**6.04e+04±2.78e+04**	6.51e+04±2.44e+04
$F_{CEC2017}19$	2.11e+03±4.65e+01	2.13e+03±9.99e+01	2.10e+03±5.44e+01	2.12e+03±6.34e+01	2.11e+03±5.34e+01	**2.09e+03±4.32e+01**
$F_{CEC2017}20$	2.39e+03±7.18e+01	2.37e+03±8.00e+01	**2.36e+03±9.67e+01**	2.43e+03±1.19e+02	2.38e+03±9.82e+01	2.41e+03±8.09e+01
$F_{CEC2017}21$	**2.42e+03±4.93e+01**	2.43e+03±2.29e+01	2.44e+03±2.33e+01	2.44e+03±2.97e+01	2.44e+03±1.98e+01	2.44e+03±3.09e+01
$F_{CEC2017}22$	2.36e+03±3.72e+01	**2.36e+03±2.87e+01**	2.37e+03±5.24e+01	2.49e+03±7.44e+02	2.36e+03±3.93e+01	2.39e+03±1.43e+02
$F_{CEC2017}23$	2.86e+03±4.21e+01	2.86e+03±3.17e+01	2.86e+03±3.16e+01	2.86e+03±3.66e+01	**2.85e+03±4.77e+01**	2.86e+03±3.09e+01
$F_{CEC2017}24$	**2.98e+03±2.54e+01**	2.99e+03±3.37e+01	3.00e+03±3.17e+01	2.99e+03±2.69e+01	2.98e+03±3.58e+01	3.00e+03±3.31e+01
$F_{CEC2017}25$	2.89e+03±1.04e+00	2.89e+03±7.42e-01	2.89e+03±1.38e+00	**2.89e+03±9.09e-01**	2.89e+03±1.17e+00	2.89e+03±1.31e+00
$F_{CEC2017}26$	3.51e+03±6.46e+02	3.44e+03±4.59e+02	3.60e+03±8.68e+02	3.54e+03±7.47e+02	3.68e+03±8.30e+02	**3.44e+03±3.44e+02**
$F_{CEC2017}27$	3.32e+03±2.46e+01	**3.31e+03±2.31e+01**	3.31e+03±2.19e+01	3.32e+03±2.47e+01	3.31e+03±2.16e+01	3.32e+03±1.76e+01
$F_{CEC2017}28$	3.22e+03±7.61e+00	3.22e+03±1.14e+01	3.22e+03±7.94e+00	3.22e+03±8.92e+00	**3.22e+03±7.11e+00**	3.22e+03±1.21e+01
$F_{CEC2017}29$	**3.96e+03±1.38e+02**	4.02e+03±1.42e+02	4.06e+03±1.30e+02	3.97e+03±1.37e+02	3.99e+03±1.18e+02	3.97e+03±1.18e+02
$F_{CEC2017}30$	6.71e+04±2.12e+04	**5.98e+04±2.26e+04**	6.79e+04±1.94e+04	6.12e+04±1.82e+04	6.71e+04±2.18e+04	6.87e+04±3.54e+04

Table 5.18: Comparison with the CSs on the benchmark functions from CEC'2011

Function	CS	PSOCS	MCS	SFCS
	Average±Std	Average±Std	Average±Std	Average±Std
$F_{CEC2011}1$	9.68e+00±4.05e+00	2.32e+01±2.38e+00	2.18e+01±4.74e+00	**6.44e+00±5.54e+00**
$F_{CEC2011}2$	-8.93e+00±8.21e-01	-4.38e+00±9.03e-01	**-1.53e+01±3.29e+00**	-1.07e+01±1.45e+00
$F_{CEC2011}3$	1.15e-05±1.44e-19	1.15e-05±4.35e-14	1.15e-05±5.85e-19	**1.15e-05±1.63e-19**
$F_{CEC2011}4$	1.40e+01±3.03e-01	1.46e+01±7.28e-01	1.92e+01±2.59e+00	**1.39e+01±2.08e-01**
$F_{CEC2011}5$	-3.08e+01±1.74e+00	-1.89e+01±1.98e+00	-3.04e+01±3.27e+00	**-3.30e+01±1.56e+00**
$F_{CEC2011}6$	-2.48e+01±1.90e+00	-1.11e+01±1.77e+00	-2.29e+01±3.20e+00	**-2.54e+01±2.15e+00**
$F_{CEC2011}7$	1.22e+00±6.66e-02	1.91e+00±1.42e-01	1.38e+00±1.62e-01	**1.12e+00±9.99e-02**
$F_{CEC2011}8$	2.20e+02±0.00e+00	2.42e+02±1.56e+01	2.42e+02±4.55e+01	**2.20e+02±0.00e+00**
$F_{CEC2011}9$	7.20e+05±4.11e+04	2.64e+06±1.06e+05	**2.85e+05±4.25e+04**	6.69e+05±4.37e+04
$F_{CEC2011}10$	-1.90e+01±1.18e+00	-8.46e+00±1.15e+00	-1.44e+01±2.74e+00	**-2.01e+01±1.25e+00**
$F_{CEC2011}11$	5.87e+05±1.08e+05	2.65e+06±6.54e+05	**7.84e+04±1.49e+04**	4.65e+05±1.05e+05
$F_{CEC2011}12$	1.61e+07±6.57e+05	4.97e+07±7.92e+05	2.56e+07±3.74e+05	**1.53e+07±4.85e+05**
$F_{CEC2011}13$	1.54e+04±1.63e+00	1.60e+04±7.10e+02	1.55e+04±2.78e+01	**1.54e+04±1.39e+00**
$F_{CEC2011}14$	**1.89e+04±8.18e+01**	2.02e+04±1.32e+03	1.92e+04±2.16e+02	1.90e+04±9.82e+01
$F_{CEC2011}15$	**3.30e+04±4.55e+01**	2.25e+05±5.96e+04	3.31e+04±7.01e+01	3.30e+04±4.25e+01
$F_{CEC2011}16$	2.14e+05±5.54e+04	3.54e+06±7.11e+06	**1.44e+05±5.16e+03**	2.12e+05±4.57e+04
$F_{CEC2011}17$	5.91e+06±1.01e+07	8.52e+09±1.37e+09	2.75e+06±5.44e+05	**2.52e+06±8.45e+05**
$F_{CEC2011}18$	1.52e+06±3.10e+05	8.65e+07±1.21e+07	**1.25e+06±4.73e+05**	1.41e+06±1.11e+05
$F_{CEC2011}19$	2.27e+06±3.29e+05	8.38e+07±1.46e+07	**1.94e+06±3.17e+05**	2.14e+06±1.89e+05
$F_{CEC2011}20$	1.49e+06±1.56e+05	8.83e+07±1.12e+07	**1.18e+06±3.36e+05**	1.38e+06±1.25e+05
$F_{CEC2011}21$	2.33e+01±2.31e+00	4.86e+01±5.96e+00	2.17e+01±3.58e+00	**1.96e+01±2.41e+00**
$w/t/l$	11/10/0	21/0/0	14/0/7	-

Table 5.19: Comparison of SFCS with the five metaheuristic algorithms on the benchmark functions from CEC'2011

Function	DE	FPA	GA	GSA	SMS	SFCS
	Average±Std	Average±Std	Average±Std	Average±Std	Average±Std	Average±Std
$F_{CEC2011}1$	**1.66e+00±4.39e+00**	1.55e+01±2.72e+00	1.97e+01±3.48e+00	2.50e+01±1.62e+00	2.27e+01±4.57e+00	6.44e+00±5.54e+00
$F_{CEC2011}2$	-4.13e+00±5.28e-01	-8.09e+00±1.05e+00	-1.00e+01±2.09e+00	-1.44e+01±3.76e+00	**-2.48e+01±1.74e+00**	-1.07e+01±1.45e+00
$F_{CEC2011}3$	1.15e-05±2.40e-19	1.15e-05±1.37e-19	1.15e-05±2.24e-12	1.15e-05±2.82e-19	1.15e-05±4.18e-15	**1.15e-05±1.63e-19**
$F_{CEC2011}4$	1.85e+01±3.24e+00	**1.38e+01±1.39e-02**	1.61e+01±2.14e+00	1.51e+01±1.25e+00	1.39e+01±9.66e-02	1.39e+01±2.08e-01
$F_{CEC2011}5$	-1.82e+01±1.22e+00	-2.90e+01±1.95e+00	-2.74e+01±2.69e+00	-2.88e+01±3.79e+00	-3.01e+01±2.95e+00	**-3.30e+01±1.56e+00**
$F_{CEC2011}6$	-1.32e+01±1.91e+00	-2.31e+01±8.43e-01	-2.08e+01±3.03e+00	-1.82e+01±3.04e+00	-2.12e+01±3.35e+00	**-2.54e+01±2.15e+00**
$F_{CEC2011}7$	1.79e+00±1.21e-01	1.33e+00±5.32e-02	1.58e+00±1.94e-01	**9.66e-01±1.83e-01**	1.40e+00±2.18e-01	1.12e+00±9.99e-02
$F_{CEC2011}8$	2.20e+02±0.00e+00	2.20e+02±0.00e+00	2.27e+02±2.34e+01	2.64e+02±3.52e+01	2.87e+03±1.28e+03	**2.20e+02±0.00e+00**
$F_{CEC2011}9$	2.51e+06±8.86e+04	1.87e+06±4.68e+04	1.05e+06±8.15e+04	9.88e+05±1.11e+05	2.35e+06±1.24e+05	**6.69e+05±4.37e+04**
$F_{CEC2011}10$	**-2.17e+01±1.08e-01**	-1.79e+01±1.83e+00	-1.16e+01±2.59e+00	-1.40e+01±2.75e-01	-1.05e+01±3.13e-01	-2.01e+01±1.25e+00
$F_{CEC2011}11$	2.09e+08±1.35e+07	1.38e+06±1.38e+05	4.72e+07±1.08e+07	7.63e+05±1.20e+05	1.00e+08±0.00e+00	**4.65e+05±1.05e+05**
$F_{CEC2011}12$	5.29e+07±7.96e+05	3.55e+07±7.92e+05	3.72e+07±1.01e+06	4.42e+07±5.72e+05	5.42e+07±8.06e+05	**1.53e+07±4.85e+05**
$F_{CEC2011}13$	**1.54e+04±1.83e-05**	1.54e+04±1.36e+00	1.56e+04±8.58e+01	1.38e+05±6.26e+04	1.60e+04±5.61e+02	1.54e+04±1.39e+00
$F_{CEC2011}14$	**1.85e+04±1.28e+02**	1.90e+04±1.50e+02	2.21e+04±2.41e+03	1.92e+04±1.43e+02	2.90e+05±2.32e+05	1.90e+04±9.82e+01
$F_{CEC2011}15$	**3.29e+04±4.23e+01**	3.30e+04±4.66e+01	3.36e+04±1.58e+03	1.51e+05±4.66e+04	2.12e+05±5.89e+04	3.30e+04±4.25e+01
$F_{CEC2011}16$	1.83e+06±1.38e+06	**1.42e+05±5.54e+03**	1.58e+05±1.16e+04	1.76e+05±3.41e+04	9.17e+06±4.95e+06	2.12e+05±4.57e+04
$F_{CEC2011}17$	6.00e+09±9.99e+08	7.02e+06±2.31e+07	5.19e+08±3.33e+08	1.33e+09±7.60e+08	1.00e+08±0.00e+00	**2.52e+06±8.45e+05**
$F_{CEC2011}18$	8.27e+07±6.73e+06	1.63e+07±5.45e+06	1.54e+07±5.50e+06	1.84e+06±1.43e+06	9.45e+07±5.52e+06	**1.41e+06±1.11e+05**
$F_{CEC2011}19$	8.73e+07±5.96e+06	1.72e+07±6.06e+06	1.59e+07±5.18e+06	3.71e+06±2.42e+06	9.56e+07±6.34e+06	**2.14e+06±1.89e+05**
$F_{CEC2011}20$	8.38e+07±6.62e+06	1.48e+07±5.27e+06	1.82e+07±4.54e+06	2.29e+06±1.75e+06	9.55e+07±7.09e+06	**1.38e+06±1.25e+05**
$F_{CEC2011}21$	4.13e+01±5.62e+00	2.69e+01±2.64e+00	3.45e+01±8.41e+00	4.80e+01±5.59e+00	4.42e+01±7.07e+00	**1.96e+01±2.41e+00**
$w/t/l$	15/1/5	15/5/1	19/1/1	15/2/4	20/0/1	-

Table 5.20: Comparison of SFCS with the SFIPSO on CEC'2013

Function	D = 10		D = 30		D = 50	
	SFIPSO Average±Std	SFCS Average±Std	SFIPSO Average±Std	SFCS Average±Std	SFIPSO Average±Std	SFCS Average±Std
$F_{CEC2013}1$	-1.39e+03±9.51e+00	**-1.40e+03±1.03e-13**	1.79e+03±9.89e+02	**-1.40e+03±5.95e-03**	8.88e+03±2.10e+03	**-1.40e+03±1.56e+00**
$F_{CEC2013}2$	2.95e+05±2.65e+05	**3.81e+03±2.93e+03**	4.49e+07±1.43e+07	**8.87e+06±2.37e+06**	8.53e+07±1.89e+07	**2.79e+07±5.11e+06**
$F_{CEC2013}3$	3.29e+08±1.90e+08	**7.49e+04±1.00e+05**	3.21e+10±8.55e+09	**1.47e+09±6.20e+08**	4.15e+10±9.78e+09	**1.27e+10±3.92e+09**
$F_{CEC2013}4$	**-4.42e+02±5.40e+02**	-3.18e+02±2.93e+02	**2.58e+04±5.56e+03**	6.38e+04±8.15e+03	**4.70e+04±7.09e+03**	1.13e+05±1.23e+04
$F_{CEC2013}5$	-9.93e+02±6.76e+00	**-1.00e+03±3.99e-09**	-3.88e+01±1.50e+02	**-1.00e+03±3.29e-02**	1.62e+02±1.99e+02	**-9.93e+02±1.78e+00**
$F_{CEC2013}6$	-8.54e+02±1.40e+01	**-9.00e+02±7.63e-02**	-3.61e+02±1.30e+02	**-8.67e+02±1.47e+01**	-3.46e+02±9.72e+01	**-8.50e+02±2.53e+00**
$F_{CEC2013}7$	-7.57e+02±1.07e+01	**-7.77e+02±5.82e+00**	-6.37e+02±2.30e+01	**-7.00e+02±1.33e+01**	-6.72e+02±1.37e+01	**-6.74e+02±1.32e+01**
$F_{CEC2013}8$	-6.80e+02±6.17e-02	**-6.80e+02±7.76e-02**	-6.79e+02±4.89e-02	**-6.79e+02±4.10e-02**	**-6.79e+02±3.70e-02**	-6.79e+02±3.74e-02
$F_{CEC2013}9$	-5.95e+02±1.00e+00	**-5.95e+02±6.18e-01**	-5.60e+02±2.30e+00	**-5.69e+02±1.92e+00**	-5.31e+02±4.01e+00	**-5.40e+02±2.15e+00**
$F_{CEC2013}10$	-4.98e+02±1.83e+00	**-5.00e+02±2.36e-02**	4.22e+01±1.52e+02	**-4.99e+02±2.24e-01**	9.77e+02±2.90e+02	**-4.66e+02±1.07e+01**
$F_{CEC2013}11$	-3.84e+02±6.92e+00	**-3.94e+02±2.18e+00**	-1.68e+02±3.24e+01	**-3.01e+02±1.92e+01**	9.42e+01±3.11e+01	**-1.39e+02±2.20e+01**
$F_{CEC2013}12$	-2.82e+02±6.81e+00	**-2.83e+02±4.72e+00**	-5.96e+01±3.13e+01	**-1.21e+02±2.81e+01**	2.01e+02±3.50e+01	**1.15e+02±4.60e+01**
$F_{CEC2013}13$	-1.78e+02±6.09e+00	**-1.80e+02±7.47e+00**	6.12e+01±1.81e+01	**1.06e+01±3.14e+01**	3.39e+02±2.66e+01	**2.74e+02±3.79e+01**
$F_{CEC2013}14$	8.41e+02±1.79e+02	**1.07e+02±6.21e+01**	6.28e+03±4.39e+02	**2.68e+03±3.61e+02**	1.27e+04±7.64e+02	**6.62e+03±5.51e+02**
$F_{CEC2013}15$	1.37e+03±2.23e+02	**6.75e+02±1.49e+02**	7.63e+03±3.40e+02	**4.43e+03±3.21e+02**	1.48e+04±4.02e+02	**9.79e+03±4.74e+02**
$F_{CEC2013}16$	2.01e+02±2.31e-01	**2.01e+02±1.37e-01**	2.03e+02±3.22e-01	**2.02e+02±3.53e-01**	2.04e+02±3.06e-01	**2.04e+02±3.19e-01**
$F_{CEC2013}17$	3.24e+02±2.97e+00	**3.21e+02±4.08e+00**	5.26e+02±2.80e+01	**4.70e+02±1.72e+01**	8.37e+02±4.41e+01	**7.16e+02±3.42e+01**
$F_{CEC2013}18$	**4.27e+02±4.98e+00**	4.29e+02±5.13e+00	6.38e+02±1.76e+01	**6.16e+02±1.94e+01**	9.49e+02±4.94e+01	**9.02e+02±3.82e+01**
$F_{CEC2013}19$	5.01e+02±3.66e-01	**5.01e+02±2.50e-01**	8.48e+02±1.84e+02	**5.12e+02±1.74e+00**	3.64e+03±1.52e+03	**5.40e+02±5.27e+00**
$F_{CEC2013}20$	6.04e+02±1.88e-01	**6.03e+02±2.41e-01**	6.15e+02±9.76e-02	**6.14e+02±4.56e-01**	6.24e+02±4.42e-01	**6.24e+02±4.54e-01**
$F_{CEC2013}21$	1.10e+03±2.72e-04	**8.43e+02±5.68e+01**	1.91e+03±2.00e+02	**9.31e+02±1.79e+01**	3.77e+03±2.13e+02	**1.16e+03±1.27e+02**
$F_{CEC2013}22$	2.27e+03±2.61e+02	**1.17e+03±9.69e+01**	7.78e+03±6.70e+02	**4.08e+03±4.16e+02**	1.52e+04±7.57e+02	**9.06e+03±5.77e+02**
$F_{CEC2013}23$	2.49e+03±2.43e+02	**1.82e+03±1.94e+02**	9.02e+03±3.42e+02	**6.03e+03±3.56e+02**	1.67e+04±3.70e+02	**1.25e+04±5.87e+02**
$F_{CEC2013}24$	1.22e+03±1.20e+01	**1.14e+03±1.40e+01**	1.31e+03±9.73e+00	**1.28e+03±8.53e+00**	1.41e+03±1.99e+01	**1.35e+03±1.08e+01**
$F_{CEC2013}25$	1.30e+03±1.42e+01	**1.28e+03±2.95e+01**	1.46e+03±6.19e+00	**1.42e+03±5.96e+00**	1.63e+03±8.78e+00	**1.55e+03±1.03e+01**
$F_{CEC2013}26$	1.37e+03±1.97e+01	**1.32e+03±7.27e+00**	1.40e+03±9.78e-01	**1.40e+03±1.81e-01**	1.56e+03±8.43e+01	**1.40e+03±7.20e-01**
$F_{CEC2013}27$	1.72e+03±2.91e+01	**1.68e+03±2.23e+01**	2.48e+03±8.98e+01	**2.40e+03±1.96e+02**	3.49e+03±1.24e+02	**3.33e+03±7.19e+01**
$F_{CEC2013}28$	1.79e+03±4.41e+01	**1.52e+03±6.10e+01**	3.49e+03±2.20e+02	**1.80e+03±4.57e+01**	6.11e+03±3.28e+02	**1.85e+03±1.07e+01**
$w/t/l$	23/4/1	-	25/2/1	-	24/3/1	-

Table 5.21: Comparison of SFCS with the SFIPSO on CEC'2017

Function	D = 10		D = 30		D = 50	
	SFIPSO Average±Std	SFCS Average±Std	SFIPSO Average±Std	SFCS Average±Std	SFIPSO Average±Std	SFCS Average±Std
$F_{CEC2017}1$	5.71e+07±5.54e+07	**1.01e+02±4.54e-01**	3.28e+09±9.61e+08	**2.42e+05±1.04e+05**	1.37e+10±3.63e+09	**1.93e+07±9.10e+06**
$F_{CEC2017}2$	4.19e+03±5.92e+03	**2.00e+02±3.02e-07**	2.15e+30±5.76e+30	**2.65e+16±6.50e+16**	3.62e+55±1.75e+56	**3.84e+41±9.66e+41**
$F_{CEC2017}3$	3.03e+02±4.12e+00	**3.00e+02±1.71e-03**	**2.35e+04±5.45e+03**	6.91e+04±1.11e+04	**1.06e+05±1.55e+04**	2.05e+05±2.26e+04
$F_{CEC2017}4$	4.22e+02±9.07e+00	**4.00e+02±1.16e-01**	1.23e+03±2.30e+02	**4.73e+02±1.97e+01**	3.18e+03±5.05e+02	**5.84e+02±3.35e+01**
$F_{CEC2017}5$	5.27e+02±7.09e+00	**5.13e+02±4.07e+00**	6.70e+02±2.05e+01	**6.35e+02±1.71e+01**	8.71e+02±2.89e+01	**8.05e+02±2.96e+01**
$F_{CEC2017}6$	6.04e+02±1.08e+00	**6.03e+02±1.51e+00**	**6.29e+02±4.41e+00**	6.38e+02±6.06e+00	**6.45e+02±3.74e+00**	6.55e+02±5.94e+00
$F_{CEC2017}7$	**7.22e+02±4.98e+00**	7.26e+02±4.11e+00	9.20e+02±2.77e+01	**8.64e+02±2.42e+01**	1.23e+03±4.76e+01	**1.12e+03±4.10e+01**
$F_{CEC2017}8$	**8.09e+02±3.30e+00**	8.13e+02±3.28e+00	9.49e+02±1.21e+01	**9.19e+02±1.54e+01**	1.18e+03±2.38e+01	**1.10e+03±3.75e+01**
$F_{CEC2017}9$	**9.03e+02±2.88e+00**	9.17e+02±1.14e+01	**1.88e+03±3.18e+02**	4.40e+03±9.30e+02	**1.03e+04±1.63e+03**	1.78e+04±3.07e+03
$F_{CEC2017}10$	2.14e+03±2.01e+02	**1.49e+03±1.16e+02**	8.26e+03±4.05e+02	**4.41e+03±3.24e+02**	1.47e+04±3.99e+02	**8.17e+03±5.18e+02**
$F_{CEC2017}11$	1.11e+03±4.96e+00	**1.10e+03±1.10e+00**	1.42e+03±6.31e+01	**1.20e+03±1.93e+01**	3.16e+03±5.61e+02	**1.41e+03±3.15e+01**
$F_{CEC2017}12$	8.33e+03±8.81e+03	**1.45e+03±7.94e+01**	1.18e+08±4.78e+07	**3.21e+05±1.38e+05**	4.27e+09±1.20e+09	**6.99e+06±2.34e+06**
$F_{CEC2017}13$	1.42e+03±8.51e+01	**1.31e+03±3.05e+00**	4.49e+06±7.63e+06	**6.45e+03±1.19e+03**	5.99e+08±3.69e+08	**6.72e+04±2.19e+04**
$F_{CEC2017}14$	1.43e+03±7.91e+00	**1.40e+03±2.05e+00**	9.41e+03±2.08e+04	**1.50e+03±1.32e+01**	7.05e+05±7.59e+05	**6.32e+03±1.76e+03**
$F_{CEC2017}15$	1.52e+03±8.19e+00	**1.50e+03±6.29e-01**	4.72e+03±2.89e+03	**1.83e+03±6.82e+01**	3.73e+05±6.49e+05	**6.60e+03±1.20e+03**
$F_{CEC2017}16$	1.68e+03±4.22e+01	**1.60e+03±1.16e+00**	3.70e+03±2.73e+02	**2.50e+03±1.39e+02**	4.83e+03±5.57e+02	**3.36e+03±2.81e+02**
$F_{CEC2017}17$	1.74e+03±6.42e+00	**1.72e+03±7.03e+00**	2.29e+03±1.50e+02	**1.93e+03±9.62e+01**	3.22e+03±2.62e+02	**2.94e+03±1.29e+02**
$F_{CEC2017}18$	8.53e+03±2.30e+04	**1.80e+03±1.88e+00**	5.60e+04±3.63e+04	**4.25e+04±1.24e+04**	2.30e+06±1.43e+06	**6.28e+05±2.53e+05**
$F_{CEC2017}19$	1.92e+03±1.43e+01	**1.90e+03±4.31e-01**	5.41e+03±7.00e+03	**1.98e+03±1.35e+01**	1.19e+05±2.01e+05	**6.46e+03±2.01e+03**
$F_{CEC2017}20$	2.06e+03±1.28e+01	**2.01e+03±7.78e+00**	2.51e+03±1.33e+02	**2.37e+03±9.85e+01**	3.73e+03±2.95e+02	**3.02e+03±1.69e+02**
$F_{CEC2017}21$	2.20e+03±2.40e+00	**2.20e+03±1.19e+00**	2.49e+03±2.00e+01	**2.42e+03±3.94e+01**	2.69e+03±3.22e+01	**2.59e+03±3.78e+01**
$F_{CEC2017}22$	2.31e+03±6.98e+00	**2.26e+03±3.27e+01**	2.89e+03±1.17e+02	**2.33e+03±1.04e+01**	1.53e+04±2.06e+03	**9.86e+03±4.62e+02**
$F_{CEC2017}23$	2.66e+03±3.15e+01	**2.62e+03±5.29e+00**	3.11e+03±4.45e+01	**2.83e+03±3.63e+01**	3.72e+03±8.73e+01	**3.18e+03±6.25e+01**
$F_{CEC2017}24$	2.62e+03±5.64e+01	**2.49e+03±2.85e+01**	3.26e+03±5.87e+01	**2.95e+03±9.08e+01**	3.87e+03±1.06e+02	**3.30e+03±6.01e+01**
$F_{CEC2017}25$	2.93e+03±1.81e+01	**2.79e+03±1.25e+02**	3.06e+03±3.45e+01	**2.89e+03±7.40e-01**	4.55e+03±2.72e+02	**3.08e+03±2.25e+01**
$F_{CEC2017}26$	3.01e+03±3.71e+01	**2.67e+03±9.59e+01**	6.66e+03±7.05e+02	**3.13e+03±2.10e+02**	1.05e+04±5.23e+02	**6.42e+03±1.40e+03**
$F_{CEC2017}27$	3.13e+03±7.32e+00	**3.09e+03±2.26e+00**	3.69e+03±7.14e+01	**3.29e+03±2.19e+01**	5.27e+03±1.76e+02	**3.90e+03±8.93e+01**
$F_{CEC2017}28$	3.18e+03±5.83e+01	**3.04e+03±1.19e+02**	3.67e+03±7.78e+01	**3.22e+03±7.66e+00**	5.46e+03±2.06e+02	**3.37e+03±3.54e+01**
$F_{CEC2017}29$	3.21e+03±1.60e+01	**3.17e+03±1.21e+01**	4.44e+03±2.16e+02	**3.88e+03±9.48e+01**	6.95e+03±4.59e+02	**4.72e+03±1.89e+02**
$F_{CEC2017}30$	2.74e+05±2.79e+05	**4.51e+03±7.49e+02**	1.21e+06±1.13e+06	**5.15e+04±1.50e+04**	9.75e+07±3.04e+07	**9.89e+06±1.04e+06**
$w/t/l$	27/0/3	-	26/1/3	-	27/0/3	-

Table 5.22: Comparison of SFDE with the DE on CEC'2013

Function	D = 10		D = 30		D = 50	
	DE Average±Std	SFDE Average±Std	DE Average±Std	SFDE Average±Std	DE Average±Std	SFDE Average±Std
$F_{CEC2013}1$	-1.40e+03±0.00e+00	**-1.40e+03±0.00e+00**	-1.40e+03±3.71e-04	**-1.40e+03±8.44e-14**	-1.40e+03±9.09e-01	**-1.40e+03±6.33e-09**
$F_{CEC2013}2$	**-1.30e+03±2.64e-09**	1.62e+05±5.99e+04	**2.19e+07±1.67e+07**	4.74e+07±1.91e+07	3.21e+08±1.41e+08	**1.93e+08±4.90e+07**
$F_{CEC2013}3$	**-1.20e+03±8.39e-02**	-1.20e+03±1.67e+00	3.40e+09±2.77e+09	**9.34e+03±2.30e+04**	5.31e+10±2.46e+10	**3.45e+08±3.41e+08**
$F_{CEC2013}4$	**-1.10e+03±2.30e-11**	8.14e+02±9.77e+02	1.01e+05±1.16e+04	**6.98e+04±1.12e+04**	1.93e+05±2.21e+04	**1.49e+05±1.43e+04**
$F_{CEC2013}5$	-1.00e+03±0.00e+00	**-1.00e+03±0.00e+00**	-1.00e+03±3.88e-03	**-1.00e+03±4.48e-12**	-9.97e+02±8.37e-01	**-1.00e+03±2.22e-05**
$F_{CEC2013}6$	**-8.94e+02±4.03e+00**	-8.92e+02±1.56e-04	-8.68e+02±1.85e+01	**-8.80e+02±1.06e+01**	-8.52e+02±1.29e+00	**-8.55e+02±1.24e+00**
$F_{CEC2013}7$	-8.00e+02±4.91e-03	**-8.00e+02±4.52e-06**	-6.95e+02±3.11e+01	**-7.94e+02±3.26e+00**	-6.26e+02±4.07e+01	**-7.51e+02±9.97e+00**
$F_{CEC2013}8$	**-6.80e+02±7.98e-02**	-6.80e+02±9.39e-02	-6.79e+02±3.90e-02	**-6.79e+02±4.78e-02**	-6.79e+02±3.24e-02	**-6.79e+02±3.87e-02**
$F_{CEC2013}9$	-5.92e+02±2.46e+00	**-5.93e+02±3.34e+00**	**-5.60e+02±1.32e+00**	-5.60e+02±1.21e+00	**-5.25e+02±1.76e+00**	-5.25e+02±1.45e+00
$F_{CEC2013}10$	**-5.00e+02±1.89e-01**	-4.99e+02±7.59e-02	-4.99e+02±2.14e-01	**-5.00e+02±9.59e-03**	-4.29e+02±2.72e+01	**-4.99e+02±2.81e-01**
$F_{CEC2013}11$	**-3.93e+02±7.12e+00**	-3.81e+02±3.93e+00	**-2.22e+02±2.35e+01**	-2.17e+02±1.02e+01	-1.36e+01±3.13e+01	**-2.73e+01±1.66e+01**
$F_{CEC2013}12$	**-2.77e+02±1.04e+01**	-2.75e+02±4.75e+00	-8.23e+01±1.49e+01	**-1.02e+02±1.16e+01**	1.44e+02±1.76e+01	**1.01e+02±1.54e+01**
$F_{CEC2013}13$	**-1.79e+02±1.01e+01**	-1.73e+02±4.48e+00	1.95e+01±1.45e+01	**2.20e+00±9.31e+00**	2.46e+02±1.92e+01	**2.06e+02±1.52e+01**
$F_{CEC2013}14$	**8.80e+02±2.78e+02**	1.06e+03±1.25e+02	6.97e+03±2.47e+02	**6.86e+03±3.23e+02**	1.32e+04±4.17e+02	**1.31e+04±3.18e+02**
$F_{CEC2013}15$	1.59e+03±2.26e+02	**1.54e+03±1.26e+02**	7.93e+03±2.87e+02	**7.90e+03±2.25e+02**	1.49e+04±3.32e+02	**1.48e+04±3.76e+02**
$F_{CEC2013}16$	**2.01e+02±1.73e-01**	2.01e+02±1.86e-01	**2.03e+02±4.01e-01**	2.03e+02±3.31e-01	2.04e+02±3.29e-01	**2.04e+02±3.26e-01**
$F_{CEC2013}17$	**3.24e+02±8.61e+00**	3.32e+02±3.52e+00	5.21e+02±2.31e+01	**5.17e+02±1.10e+01**	7.52e+02±2.71e+01	**7.33e+02±1.48e+01**
$F_{CEC2013}18$	4.39e+02±5.10e+00	**4.37e+02±4.59e+00**	6.50e+02±1.34e+01	**6.30e+02±1.11e+01**	8.88e+02±1.90e+01	**8.51e+02±1.44e+01**
$F_{CEC2013}19$	**5.01e+02±7.95e-01**	5.02e+02±3.30e-01	5.18e+02±1.18e+00	**5.17e+02±7.10e-01**	5.41e+02±4.04e+00	**5.34e+02±1.43e+00**
$F_{CEC2013}20$	6.03e+02±3.12e-01	**6.02e+02±2.13e-01**	6.14e+02±2.82e-01	**6.13e+02±2.77e-01**	6.25e+02±1.20e-01	**6.23e+02±2.90e-01**
$F_{CEC2013}21$	**1.09e+03±5.08e+01**	1.10e+03±4.63e-13	1.07e+03±8.82e+01	**1.05e+03±9.39e+01**	**1.22e+03±4.16e+02**	1.27e+03±4.59e+02
$F_{CEC2013}22$	**2.02e+03±2.90e+02**	2.07e+03±1.50e+02	8.38e+03±3.16e+02	**8.27e+03±3.21e+02**	1.51e+04±3.88e+02	**1.47e+04±4.86e+02**
$F_{CEC2013}23$	**2.55e+03±1.76e+02**	2.61e+03±1.39e+02	9.19e+03±3.11e+02	**9.16e+03±2.68e+02**	1.62e+04±4.18e+02	**1.61e+04±3.84e+02**
$F_{CEC2013}24$	1.21e+03±1.01e+01	**1.21e+03±1.03e+01**	1.32e+03±3.94e+00	**1.31e+03±2.79e+00**	1.41e+03±5.16e+00	**1.41e+03±5.47e+00**
$F_{CEC2013}25$	1.32e+03±3.09e+00	**1.32e+03±4.87e+00**	1.42e+03±3.90e+00	**1.42e+03±3.29e+00**	1.53e+03±5.79e+00	**1.52e+03±5.26e+00**
$F_{CEC2013}26$	**1.32e+03±7.58e+00**	1.33e+03±1.41e+01	1.40e+03±2.50e+00	**1.40e+03±1.13e+00**	**1.68e+03±3.43e+01**	1.69e+03±1.99e+01
$F_{CEC2013}27$	1.60e+03±2.29e-03	**1.60e+03±0.00e+00**	2.70e+03±2.59e+01	**2.64e+03±4.25e+01**	3.64e+03±6.03e+01	**3.59e+03±4.82e+01**
$F_{CEC2013}28$	**1.69e+03±5.07e+01**	1.70e+03±0.00e+00	1.70e+03±1.07e+00	**1.70e+03±1.18e-07**	1.82e+03±6.42e+00	**1.80e+03±6.67e-04**
$w/t/l$	6/13/9	-	16/11/1	-	22/6/0	-

Table 5.23: Comparison of SFDE with the DE on CEC'2017

Function	D = 10		D = 30		D = 50	
	DE Average±Std	SFDE Average±Std	DE Average±Std	SFDE Average±Std	DE Average±Std	SFDE Average±Std
$F_{CEC2017}1$	**1.00e+02±0.00e+00**	2.02e+03±1.22e+03	2.14e+04±1.55e+04	**4.72e+03±5.17e+03**	5.35e+06±3.36e+06	**4.36e+03±4.14e+03**
$F_{CEC2017}2$	**2.00e+02±0.00e+00**	2.00e+02±4.48e-06	**5.11e+23±1.91e+24**	7.32e+23±1.88e+24	1.59e+64±8.70e+64	**1.62e+57±8.59e+57**
$F_{CEC2017}3$	3.00e+02±0.00e+00	**3.00e+02±0.00e+00**	1.40e+05±2.04e+04	**8.36e+04±1.92e+04**	3.55e+05±3.64e+04	**2.77e+05±3.72e+04**
$F_{CEC2017}4$	**4.00e+02±0.00e+00**	4.00e+02±6.51e-08	4.93e+02±1.75e+01	**4.66e+02±1.73e+01**	6.19e+02±3.35e+01	**5.46e+02±5.33e+01**
$F_{CEC2017}5$	**5.15e+02±1.06e+01**	5.25e+02±4.50e+00	7.17e+02±1.17e+01	**6.95e+02±1.07e+01**	9.40e+02±2.07e+01	**9.02e+02±1.44e+01**
$F_{CEC2017}6$	6.00e+02±1.55e-10	**6.00e+02±0.00e+00**	6.01e+02±3.12e-01	**6.00e+02±5.88e-05**	6.07e+02±1.90e+00	**6.00e+02±9.45e-03**
$F_{CEC2017}7$	**7.28e+02±9.39e+00**	7.36e+02±3.53e+00	9.53e+02±1.05e+01	**9.27e+02±1.13e+01**	1.19e+03±1.62e+01	**1.15e+03±1.69e+01**
$F_{CEC2017}8$	**8.19e+02±9.16e+00**	8.25e+02±4.90e+00	1.02e+03±1.48e+01	**1.00e+03±1.17e+01**	1.25e+03±2.00e+01	**1.20e+03±1.77e+01**
$F_{CEC2017}9$	9.00e+02±0.00e+00	**9.00e+02±0.00e+00**	9.02e+02±1.14e+00	**9.00e+02±2.09e-12**	1.38e+03±2.83e+02	**9.00e+02±1.59e-01**
$F_{CEC2017}10$	2.23e+03±1.65e+02	**2.21e+03±1.27e+02**	**8.56e+03±3.47e+02**	8.59e+03±2.34e+02	1.52e+04±3.80e+02	**1.50e+04±3.18e+02**
$F_{CEC2017}11$	**1.10e+03±7.87e-01**	1.10e+03±2.24e+00	1.20e+03±2.50e+01	**1.19e+03±3.03e+01**	1.36e+03±2.83e+01	**1.34e+03±1.99e+01**
$F_{CEC2017}12$	**1.20e+03±3.33e+00**	2.49e+04±1.77e+04	**2.93e+05±2.76e+05**	3.79e+06±2.99e+06	**1.36e+07±6.51e+06**	1.54e+07±7.03e+06
$F_{CEC2017}13$	**1.30e+03±2.39e+00**	1.34e+03±9.35e+00	**1.57e+03±5.82e+01**	1.30e+05±5.33e+04	**2.19e+04±6.78e+03**	4.11e+05±4.11e+05
$F_{CEC2017}14$	**1.40e+03±7.24e-01**	1.43e+03±2.83e+00	**1.48e+03±6.83e+00**	2.69e+03±7.14e+02	**1.60e+03±1.60e+01**	1.06e+05±4.44e+04
$F_{CEC2017}15$	**1.50e+03±5.22e-01**	1.51e+03±2.20e+00	**1.57e+03±1.30e+01**	1.89e+04±8.02e+03	**2.00e+03±9.60e+01**	1.20e+05±6.73e+04
$F_{CEC2017}16$	**1.60e+03±2.34e-01**	1.60e+03±3.15e-01	3.40e+03±1.27e+02	**3.20e+03±1.43e+02**	5.40e+03±2.38e+02	**5.11e+03±1.77e+02**
$F_{CEC2017}17$	**1.70e+03±2.98e+00**	1.72e+03±9.70e+00	**2.14e+03±2.23e+02**	2.28e+03±9.03e+01	4.15e+03±2.17e+02	**3.95e+03±1.70e+02**
$F_{CEC2017}18$	**1.80e+03±3.83e-01**	1.89e+03±3.67e+01	**1.88e+03±1.45e+01**	9.14e+05±4.51e+05	**6.39e+04±5.42e+04**	6.77e+06±4.12e+06
$F_{CEC2017}19$	**1.90e+03±2.52e-01**	1.91e+03±1.23e+00	**1.94e+03±5.50e+00**	2.43e+04±1.55e+04	**2.03e+03±1.53e+01**	6.55e+04±3.66e+04
$F_{CEC2017}20$	2.00e+03±3.82e-01	**2.00e+03±2.24e-01**	**2.66e+03±2.32e+02**	2.68e+03±1.35e+02	4.24e+03±1.88e+02	**4.08e+03±1.83e+02**
$F_{CEC2017}21$	**2.31e+03±3.72e+01**	2.32e+03±3.92e+01	2.52e+03±1.37e+01	**2.49e+03±1.30e+01**	2.74e+03±1.87e+01	**2.70e+03±1.99e+01**
$F_{CEC2017}22$	**2.28e+03±3.84e+01**	2.30e+03±1.87e+01	8.51e+03±2.64e+03	**7.61e+03±3.30e+03**	1.64e+04±3.34e+02	**1.62e+04±4.37e+02**
$F_{CEC2017}23$	**2.61e+03±5.65e+00**	2.62e+03±8.97e+00	2.88e+03±1.83e+01	**2.85e+03±1.21e+01**	3.27e+03±3.28e+01	**3.15e+03±2.14e+01**
$F_{CEC2017}24$	**2.72e+03±7.36e+01**	2.75e+03±1.13e+01	3.06e+03±2.03e+01	**3.02e+03±1.07e+01**	3.49e+03±5.18e+01	**3.31e+03±1.92e+01**
$F_{CEC2017}25$	**2.90e+03±1.15e+01**	2.91e+03±2.19e+01	2.88e+03±1.01e+00	**2.88e+03±5.16e-02**	3.13e+03±4.47e+01	**2.99e+03±1.80e+01**
$F_{CEC2017}26$	2.90e+03±0.00e+00	**2.90e+03±0.00e+00**	5.68e+03±1.55e+02	**5.39e+03±1.47e+02**	8.40e+03±2.24e+02	**7.56e+03±1.70e+02**
$F_{CEC2017}27$	3.08e+03±3.49e-01	**3.08e+03±2.35e-01**	3.30e+03±1.81e+01	**3.26e+03±1.53e+01**	4.16e+03±1.21e+02	**3.93e+03±8.94e+01**
$F_{CEC2017}28$	3.32e+03±4.26e+00	**3.32e+03±8.01e+00**	3.22e+03±1.31e+01	**3.16e+03±5.71e+01**	3.49e+03±8.30e+01	**3.28e+03±2.00e+01**
$F_{CEC2017}29$	**3.13e+03±6.42e+00**	3.18e+03±6.90e+00	4.46e+03±1.59e+02	**4.26e+03±1.90e+02**	5.96e+03±2.90e+02	**5.48e+03±2.00e+02**
$F_{CEC2017}30$	**3.95e+03±3.96e+02**	1.41e+04±8.75e+03	**3.47e+04±1.84e+04**	3.12e+05±1.61e+05	**6.81e+06±1.74e+06**	2.39e+07±6.42e+06
$w/t/l$	3/6/21	-	18/3/9	-	23/1/6	-

Table 5.24: Comparison of SFFA with the FA on CEC'2013

Function	D = 10		D = 30		D = 50	
	FA Average±Std	SFFA Average±Std	FA Average±Std	SFFA Average±Std	FA Average±Std	SFFA Average±Std
$F_{CEC2013}1$	-1.30e+03±5.58e+02	**-1.40e+03±6.18e-09**	3.53e+04±3.68e+04	**-1.40e+03±2.46e-13**	1.30e+05±4.01e+04	**-1.40e+03±1.22e-12**
$F_{CEC2013}2$	2.35e+07±2.25e+07	**4.23e+04±3.88e+04**	9.77e+08±6.78e+08	**1.83e+06±8.03e+05**	3.74e+09±1.34e+09	**5.47e+06±2.09e+06**
$F_{CEC2013}3$	3.90e+09±6.42e+09	**4.42e+06±9.35e+06**	4.69e+18±2.08e+19	**1.55e+08±2.40e+08**	6.21e+17±1.44e+18	**6.56e+08±4.16e+08**
$F_{CEC2013}4$	9.10e+04±8.73e+04	**3.83e+02±8.43e+02**	2.18e+05±1.21e+05	**2.93e+04±1.07e+04**	3.23e+05±1.09e+05	**6.25e+04±1.72e+04**
$F_{CEC2013}5$	-9.43e+02±1.78e+02	**-1.00e+03±2.17e-09**	1.77e+04±1.76e+04	**-1.00e+03±1.54e-12**	5.46e+04±2.36e+04	**-1.00e+03±8.28e-06**
$F_{CEC2013}6$	-8.63e+02±6.80e+01	**-8.95e+02±4.91e+00**	5.84e+03±6.76e+03	**-8.68e+02±2.67e+01**	1.77e+04±1.06e+04	**-8.41e+02±2.19e+01**
$F_{CEC2013}7$	-6.84e+02±5.60e+01	**-7.99e+02±1.72e+00**	1.53e+06±4.72e+06	**-7.49e+02±2.49e+01**	5.62e+05±6.87e+05	**-7.24e+02±2.02e+01**
$F_{CEC2013}8$	-6.79e+02±1.38e-01	**-6.80e+02±8.04e-02**	-6.79e+02±7.53e-02	**-6.79e+02±5.61e-02**	-6.79e+02±5.44e-02	**-6.79e+02±3.00e-02**
$F_{CEC2013}9$	-5.90e+02±2.59e+00	**-5.97e+02±1.75e+00**	-5.53e+02±1.61e+00	**-5.63e+02±7.69e+00**	-5.17e+02±1.99e+00	**-5.30e+02±1.01e+01**
$F_{CEC2013}10$	-4.68e+02±1.46e+02	**-5.00e+02±1.82e-01**	5.12e+03±4.96e+03	**-5.00e+02±1.23e-01**	1.80e+04±7.58e+03	**-4.99e+02±5.98e-01**
$F_{CEC2013}11$	-3.52e+02±2.61e+01	**-3.91e+02±4.36e+00**	2.13e+02±4.02e+02	**-3.26e+02±3.23e+01**	1.65e+03±6.11e+02	**-1.32e+02±7.65e+01**
$F_{CEC2013}12$	-2.29e+02±2.45e+01	**-2.91e+02±6.73e+00**	3.85e+02±3.50e+02	**-2.22e+02±5.10e+01**	1.71e+03±2.87e+02	**-7.47e+01±1.07e+02**
$F_{CEC2013}13$	-1.36e+02±1.35e+01	**-1.81e+02±8.40e+00**	4.83e+02±3.12e+02	**-3.92e+01±4.15e+01**	1.79e+03±3.31e+02	**1.78e+02±9.27e+01**
$F_{CEC2013}14$	1.90e+03±6.01e+02	**3.08e+02±3.16e+02**	9.09e+03±4.13e+02	**3.78e+03±2.63e+03**	1.63e+04±5.29e+02	**7.52e+03±5.02e+03**
$F_{CEC2013}15$	2.36e+03±2.12e+02	**1.51e+03±3.01e+02**	9.25e+03±4.16e+02	**7.81e+03±2.44e+02**	1.68e+04±5.55e+02	**1.48e+04±3.23e+02**
$F_{CEC2013}16$	2.03e+02±7.29e-01	**2.01e+02±2.10e-01**	2.05e+02±8.63e-01	**2.03e+02±3.24e-01**	2.06e+02±6.44e-01	**2.04e+02±3.18e-01**
$F_{CEC2013}17$	3.76e+02±1.64e+01	**3.17e+02±3.83e+00**	1.18e+03±6.63e+02	**3.93e+02±2.69e+01**	3.56e+03±1.67e+03	**5.25e+02±6.88e+01**
$F_{CEC2013}18$	4.78e+02±2.20e+01	**4.34e+02±3.07e+00**	1.26e+03±6.24e+02	**6.22e+02±1.18e+01**	3.86e+03±1.50e+03	**8.50e+02±2.24e+01**
$F_{CEC2013}19$	6.85e+02±6.85e+02	**5.01e+02±5.46e-01**	1.55e+06±1.74e+06	**5.06e+02±2.20e+00**	1.29e+07±6.14e+06	**5.21e+02±8.75e+00**
$F_{CEC2013}20$	6.04e+02±3.57e-01	**6.02e+02±6.44e-01**	6.15e+02±1.61e-01	**6.12e+02±6.81e-01**	6.25e+02±1.31e-08	**6.22e+02±8.87e-01**
$F_{CEC2013}21$	**1.09e+03±4.88e+01**	1.10e+03±1.83e+01	3.14e+03±2.17e+03	**1.02e+03±7.37e+01**	8.92e+03±4.29e+03	**1.61e+03±3.71e+02**
$F_{CEC2013}22$	3.23e+03±4.70e+02	**1.19e+03±2.32e+02**	1.06e+04±3.60e+02	**3.93e+03±1.58e+03**	1.82e+04±4.00e+02	**6.14e+03±1.05e+03**
$F_{CEC2013}23$	3.54e+03±2.48e+02	**1.64e+03±4.24e+02**	1.06e+04±3.82e+02	**8.93e+03±1.04e+03**	1.86e+04±4.02e+02	**1.66e+04±7.97e+02**
$F_{CEC2013}24$	1.23e+03±1.05e+01	**1.21e+03±5.34e+00**	1.38e+03±6.85e+01	**1.26e+03±1.18e+01**	1.74e+03±1.82e+02	**1.33e+03±2.18e+01**
$F_{CEC2013}25$	1.32e+03±1.13e+01	**1.30e+03±4.38e+00**	1.49e+03±3.71e+01	**1.39e+03±1.31e+01**	1.71e+03±3.08e+01	**1.50e+03±1.71e+01**
$F_{CEC2013}26$	1.40e+03±4.86e+01	**1.35e+03±4.41e+01**	1.59e+03±7.29e+01	**1.41e+03±3.71e+01**	1.74e+03±2.20e+01	**1.59e+03±8.66e+01**
$F_{CEC2013}27$	1.79e+03±1.27e+02	**1.61e+03±2.98e+01**	2.90e+03±1.68e+02	**2.19e+03±8.07e+01**	4.34e+03±1.74e+02	**2.92e+03±1.60e+02**
$F_{CEC2013}28$	1.89e+03±2.51e+02	**1.74e+03±1.06e+02**	7.52e+03±3.21e+03	**1.94e+03±6.72e+02**	1.49e+04±1.75e+03	**2.37e+03±1.31e+03**
$w/t/l$	28/0/0	-	28/0/0	-	28/0/0	-

Table 5.25: Comparison of SFFA with the FA on CEC'2017

Function	D = 10		D = 30		D = 50	
	FA Average±Std	SFFA Average±Std	FA Average±Std	SFFA Average±Std	FA Average±Std	SFFA Average±Std
$F_{CEC2017}1$	2.18e+07±1.14e+08	**9.81e+02±9.86e+02**	1.73e+10±3.61e+10	**4.54e+03±4.61e+03**	1.42e+11±1.04e+11	**2.48e+03±2.89e+03**
$F_{CEC2017}2$	1.93e+10±6.27e+10	**2.00e+02±3.91e-05**	1.44e+48±4.18e+48	**2.22e+18±1.22e+19**	4.19e+86±1.84e+87	**5.92e+56±3.24e+57**
$F_{CEC2017}3$	2.79e+04±1.44e+04	**3.00e+02±9.20e-14**	1.57e+06±6.28e+06	**1.59e+04±7.36e+03**	1.06e+07±4.12e+07	**1.47e+05±1.91e+04**
$F_{CEC2017}4$	4.44e+02±1.34e+02	**4.01e+02±5.54e-01**	9.31e+03±1.47e+04	**4.68e+02±3.83e+01**	7.09e+04±2.35e+04	**5.30e+02±4.80e+01**
$F_{CEC2017}5$	5.51e+02±2.33e+01	**5.09e+02±4.67e+00**	9.08e+02±1.46e+02	**5.74e+02±2.94e+01**	1.50e+03±1.77e+02	**7.04e+02±7.60e+01**
$F_{CEC2017}6$	6.07e+02±1.08e+01	**6.00e+02±4.51e-03**	6.97e+02±2.85e+01	**6.02e+02±2.52e+00**	7.34e+02±1.39e+01	**6.17e+02±1.32e+01**
$F_{CEC2017}7$	7.74e+02±2.14e+01	**7.17e+02±3.02e+00**	1.62e+03±6.08e+02	**7.91e+02±2.64e+01**	3.77e+03±1.68e+03	**9.51e+02±5.77e+01**
$F_{CEC2017}8$	8.50e+02±2.63e+01	**8.10e+02±4.54e+00**	1.16e+03±1.12e+02	**8.69e+02±2.49e+01**	1.84e+03±1.24e+02	**9.85e+02±6.00e+01**
$F_{CEC2017}9$	1.10e+03±3.80e+02	**9.00e+02±1.91e+00**	2.16e+04±7.77e+03	**9.54e+02±6.85e+01**	7.52e+04±1.47e+04	**2.10e+03±1.42e+03**
$F_{CEC2017}10$	3.13e+03±3.07e+02	**1.66e+03±4.28e+02**	1.01e+04±4.19e+02	**8.02e+03±1.41e+03**	1.71e+04±3.54e+02	**1.38e+04±2.94e+03**
$F_{CEC2017}11$	2.10e+03±1.72e+03	**1.11e+03±3.53e+00**	2.46e+04±8.81e+03	**1.15e+03±3.01e+01**	6.64e+04±1.88e+04	**1.21e+03±4.76e+01**
$F_{CEC2017}12$	8.57e+06±1.71e+07	**1.05e+04±5.85e+03**	8.29e+09±7.84e+09	**3.46e+05±4.05e+05**	8.27e+10±3.81e+10	**1.99e+06±8.42e+05**
$F_{CEC2017}13$	2.02e+05±6.10e+05	**7.54e+03±6.09e+03**	6.43e+09±7.58e+09	**1.24e+04±9.23e+03**	4.58e+10±2.19e+10	**3.99e+03±2.91e+03**
$F_{CEC2017}14$	2.65e+04±8.63e+04	**1.59e+03±2.10e+02**	1.55e+07±8.21e+06	**1.82e+05±1.85e+05**	1.24e+08±7.34e+07	**1.05e+06±1.15e+06**
$F_{CEC2017}15$	3.28e+04±2.39e+04	**3.97e+03±4.35e+03**	1.16e+09±1.26e+09	**9.28e+03±9.37e+03**	1.68e+10±7.85e+09	**5.89e+03±4.39e+03**
$F_{CEC2017}16$	2.07e+03±2.46e+02	**1.68e+03±8.08e+01**	5.85e+03±1.04e+03	**2.43e+03±3.14e+02**	1.10e+04±1.37e+03	**3.18e+03±3.90e+02**
$F_{CEC2017}17$	1.92e+03±1.32e+02	**1.72e+03±1.98e+01**	3.87e+03±1.07e+03	**2.13e+03±2.92e+02**	1.64e+05±1.53e+05	**2.99e+03±4.17e+02**
$F_{CEC2017}18$	4.18e+06±1.14e+07	**9.85e+03±8.31e+03**	2.17e+08±1.37e+08	**1.05e+06±1.25e+06**	4.69e+08±1.92e+08	**3.14e+06±2.26e+06**
$F_{CEC2017}19$	7.71e+05±3.59e+06	**5.72e+03±4.56e+03**	1.46e+09±1.62e+09	**7.07e+03±5.23e+03**	7.10e+09±2.03e+09	**1.63e+04±1.06e+04**
$F_{CEC2017}20$	2.22e+03±1.22e+02	**2.03e+03±4.50e+01**	3.47e+03±1.64e+02	**2.47e+03±2.69e+02**	5.05e+03±1.97e+02	**3.67e+03±7.96e+02**
$F_{CEC2017}21$	2.33e+03±5.45e+01	**2.22e+03±3.82e+01**	2.73e+03±1.08e+02	**2.37e+03±2.59e+01**	3.36e+03±1.31e+02	**2.47e+03±3.28e+01**
$F_{CEC2017}22$	2.31e+03±2.38e+00	**2.30e+03±1.35e+01**	1.02e+04±2.78e+03	**2.30e+03±1.18e+00**	1.87e+04±5.21e+02	**1.33e+04±5.28e+03**
$F_{CEC2017}23$	2.66e+03±2.51e+01	**2.61e+03±5.28e+00**	3.43e+03±2.56e+02	**2.77e+03±4.65e+01**	4.66e+03±2.89e+02	**3.00e+03±6.56e+01**
$F_{CEC2017}24$	2.77e+03±6.02e+01	**2.70e+03±8.18e+01**	3.72e+03±3.69e+02	**2.90e+03±3.54e+01**	5.14e+03±3.82e+02	**3.15e+03±8.39e+01**
$F_{CEC2017}25$	2.96e+03±9.77e+01	**2.92e+03±2.33e+01**	4.55e+03±3.29e+03	**2.89e+03±1.66e+01**	3.84e+04±1.34e+04	**3.07e+03±3.03e+01**
$F_{CEC2017}26$	3.25e+03±5.36e+02	**2.90e+03±1.01e+01**	9.35e+03±2.98e+03	**4.55e+03±7.38e+02**	2.41e+04±4.33e+03	**7.09e+03±1.08e+03**
$F_{CEC2017}27$	3.13e+03±5.52e+01	**3.10e+03±5.37e+00**	4.40e+03±6.67e+02	**3.25e+03±2.33e+01**	7.42e+03±5.46e+02	**3.61e+03±1.19e+02**
$F_{CEC2017}28$	3.41e+03±1.89e+02	**3.16e+03±1.21e+02**	7.85e+03±3.21e+03	**3.19e+03±3.96e+01**	1.88e+04±4.87e+03	**3.32e+03±3.11e+01**
$F_{CEC2017}29$	3.44e+03±1.56e+02	**3.17e+03±2.31e+01**	7.65e+03±2.24e+03	**3.79e+03±2.60e+02**	1.96e+05±1.92e+05	**4.39e+03±3.39e+02**
$F_{CEC2017}30$	1.24e+07±1.69e+07	**1.59e+05±3.96e+05**	7.99e+08±9.19e+08	**8.55e+03±2.74e+03**	9.94e+09±4.60e+09	**9.47e+05±1.49e+05**
$w/t/l$	30/0/0	-	29/1/0	-	30/0/0	-

Chapter 6

Conclusion

First, a novel dendritic neural model trained by the AIS algorithm, termed the AISDNM, is introduced. To evaluate the performance of the AISDNM, eight classification datasets and eight prediction problems are used in the experiments. The Taguchi method is employed to seek the most suitable user-defined parameter setting for each problem. The experimental results demonstrate that the AISDNM clearly outperforms the MLP, DT, line-SVM, rbf-SVM, poly-SVM and DNM in terms of various evaluation criteria, which suggests that the powerful search ability can present the AISDNM from trapping into the local minimum. In addition, it also has been proven that the AISDNM can remove redundant synaptic layers and useless dendritic layers to simplify the structural morphology for each classification problem. Then, the simplified structure of the AISDNM can be transformed into an LCC without sacrificing accuracy. The LCC employs the binary computation, rather than the floating-point calculation of other machine learning techniques. Thus, it is easy for hardware implementation and parallel computation. The LCC may have its own advantage to deal with big data problems due to high computation speed. Applying the AISDNM to solve more complex real-world problems is also worth investigating.

Second, we focused on the architecture design of scale-free population topology to enhance the performance of the CS algorithm. Specifically, all individuals of the population can be regarded as the nodes in the network, and each individual is restricted to exchange information with the other individuals who connect to it. The low assortativity of the scale-free architecture design can control the influence of competent individuals on the entire population, which is beneficial to maintaining better diversity and supports SFCS architecture design to fight premature convergence. Meanwhile, the power-law distribution of a scale-free population topology ensures information transmission between the competent individuals and the corrupt individuals. In other words, corrupt individuals can learn more information from competent individuals without incurring the cost of random attempts, which guarantees the exploitation capability of the SFCS architecture design. The analysis of computational complexity indicates that the introduction of the scale-free population topology into CS does not increase the computational cost. In addition, extensive comparative experiments verify the superiority of the scale-free architecture design. In our experiments, SFCS architecture design was compared with the CS algorithm, two CS variants, and five other techniques. The corresponding experimental results and statistical analysis demonstrated that SFCS architecture design performed the best on most 58 benchmark functions with $10 \sim 50$ dimensions, which suggests that the scale-free population topology enables the SFCS algorithm to allow a better trade-off between exploitation and exploration. Next, the analysis of parameter sensibility implied that the parameter M_0 has little effect on SFCS architecture design. To further validate the effectiveness of the scale-free population topology, the SFCS algorithm was also applied to 21 real-world optimization problems. Again, the comparison results showed that the SFCS algorithm can achieve superior performance in most cases compared with other excellent algorithms. Finally, we prove that our scale-free idea is more effective despite its simplicity by comparing the SFCS and SFIPSO. The effectiveness of the scale-free population topology on the DE and FA verify that it can be considered an effective tool for improving the optimization performances of the SFCS algorithm and can be extended to other excellent population-based optimization algorithms easily owing to its simple population topology structure with the ease of implementation and high computational efficiency. Furthermore, incorporating other complex networks into evolutionary algorithms is also worth further investigating.

Bibliography

[1] Bayya Yegnanarayana. *Artificial neural networks*. PHI Learning Pvt. Ltd., 2009.

[2] Warren S McCulloch and Walter Pitts. A logical calculus of the ideas immanent in nervous activity. *The bulletin of mathematical biophysics*, 5(4):115–133, 1943.

[3] Frank Rosenblatt. *The perceptron, a perceiving and recognizing automaton Project Para*. Cornell Aeronautical Laboratory, 1957.

[4] Michael London and Michael Häusser. Dendritic computation. *Annu. Rev. Neurosci.*, 28:503–532, 2005.

[5] Wilfrid Rall. Electrophysiology of a dendritic neuron model. *Biophysical journal*, 2(2):145–167, 1962.

[6] Wilfrid Rall and John Rinzel. Branch input resistance and steady attenuation for input to one branch of a dendritic neuron model. *Biophysical journal*, 13(7):648–688, 1973.

[7] John Rinzel and Wilfrid Rall. Transient response in a dendritic neuron model for current injected at one branch. *Biophysical Journal*, 14(10):759–790, 1974.

[8] Wilfrid Rall, Robert E Burke, William R Holmes, James J Jack, Stephan J Redman, and Idan Segev. Matching dendritic neuron models to experimental data. *Physiological Reviews*, 72(suppl_4):S159–S186, 1992.

[9] Wilfrid Rall. *The theoretical foundation of dendritic function: selected papers of Wilfrid Rall with commentaries*. MIT press, 1995.

[10] Cristopher M Niell, Martin P Meyer, and Stephen J Smith. In vivo imaging of synapse formation on a growing dendritic arbor. *Nature neuroscience*, 7(3):254, 2004.

[11] Christof Koch, Tomaso Poggio, and Vincent Torre. Retinal ganglion cells: a functional interpretation of dendritic morphology. *Philosophical Transactions of the Royal Society of London. B, Biological Sciences*, 298(1090):227–263, 1982.

[12] Christof Koch, Tomaso Poggio, and Vincent Torre. Nonlinear interactions in a dendritic tree: localization, timing, and role in information processing. *Proceedings of the National Academy of Sciences*, 80(9):2799–2802, 1983.

[13] W Rowland Taylor, Shigang He, William R Levick, and David I Vaney. Dendritic computation of direction selectivity by retinal ganglion cells. *Science*, 289(5488):2347–2350, 2000.

[14] Idan Segev. Sound grounds for computing dendrites. *Nature*, 393(6682):207, 1998.

[15] Robert Legenstein and Wolfgang Maass. Branch-specific plasticity enables self-organization of nonlinear computation in single neurons. *Journal of Neuroscience*, 31(30):10787–10802, 2011.

[16] Yu Xue, Pengcheng Jiang, Ferrante Neri, and Jiayu Liang. A multiobjective evolutionary approach based on graph-in-graph for neural architecture search of convolutional neural networks. *International Journal of Neural Systems*.

[17] Damien O'Neill, Bing Xue, and Mengjie Zhang. Evolutionary neural architecture search for high-dimensional skip-connection structures on densenet style networks. *IEEE Transactions on Evolutionary Computation*, 2021.

[18] Yuki Todo, Hiroki Tamura, Kazuya Yamashita, and Zheng Tang. Unsupervised learnable neuron model with nonlinear interaction on dendrites. *Neural Networks*, 60:96–103, 2014.

[19] Junkai Ji, Shangce Gao, Jiujun Cheng, Zheng Tang, and Yuki Todo. An approximate logic neuron model with a dendritic structure. *Neurocomputing*, 173:1775–1783, 2016.

[20] Cheng Tang, Junkai Ji, Yajiao Tang, Shangce Gao, Zheng Tang, and Yuki Todo. A novel machine learning technique for computer-aided diagnosis. *Engineering Applications of Artificial Intelligence*, 92:103627, 2020.

[21] Zijun Sha, Lin Hu, Yuki Todo, Junkai Ji, Shangce Gao, and Zheng Tang. A breast cancer classifier using a neuron model with dendritic nonlinearity. *IEICE TRANSACTIONS on Information and Systems*, 98(7):1365–1376, 2015.

[22] Tao Jiang, Shangce Gao, Dizhou Wang, Junkai Ji, Yuki Todo, and Zheng Tang. A neuron model with synaptic nonlinearities in a dendritic tree for liver disorders. *IEEJ Transactions on Electrical and Electronic Engineering*, 12(1):105–115, 2017.

[23] Minhui Dong, Cheng Tang, Junkai Ji, Qiuzhen Lin, and Ka-Chun Wong. Transmission trend of the covid-19 pandemic predicted by dendritic neural regression. *Applied Soft Computing*, page 107683, 2021.

[24] Zhenyu Song, Cheng Tang, Junkai Ji, Yuki Todo, and Zheng Tang. A simple dendritic neural network model-based approach for daily pm2. 5 concentration prediction. *Electronics*, 10(4):373, 2021.

[25] Tianle Zhou, Shangce Gao, Jiahai Wang, Chaoyi Chu, Yuki Todo, and Zheng Tang. Financial time series prediction using a dendritic neuron model. *Knowledge-Based Systems*, 105:214–224, 2016.

[26] David E Rumelhart, Geoffrey E Hinton, and Ronald J Williams. Learning internal representations by error propagation. Technical report, California Univ San Diego La Jolla Inst for Cognitive Science, 1985.

[27] Monica Bianchini and Marco Gori. Optimal learning in artificial neural networks: A review of theoretical results. *Neurocomputing*, 13(2-4):313–346, 1996.

[28] XuGang Wang, Zheng Tang, Hiroki Tamura, Masahiro Ishii, and WD Sun. An improved backpropagation algorithm to avoid the local minima problem. *Neurocomputing*, 56:455–460, 2004.

[29] Dipankar Dasgupta. An overview of artificial immune systems and their applications. In *Artificial immune systems and their applications*, pages 3–21. Springer, 1993.

[30] Jon Timmis. *Artificial immune systems: A novel data analysis technique inspired by the immune network theory*. PhD thesis, Department of Computer Science, 2000.

[31] Leandro Nunes Castro, Leandro Nunes De Castro, and Jonathan Timmis. *Artificial immune systems: a new computational intelligence approach*. Springer Science & Business Media, 2002.

[32] Dionisios N Sotiropoulos, George A Tsihrintzis, Anastasios Savvopoulos, and Maria Virvou. Artificial immune system-based customer data clustering in an e-shopping application. In *International Conference on Knowledge-Based and Intelligent Information and Engineering Systems*, pages 960–967. Springer, 2006.

[33] Dipankar Dasgupta, Senhua Yu, and Fernando Nino. Recent advances in artificial immune systems: models and applications. *Applied Soft Computing*, 11(2):1574–1587, 2011.

[34] Dimitrios G Giatzitzoglou, Dionisios N Sotiropoulos, and George A Tsihrintzis. Airs-x: An extension to the original artificial immune recognition learning algorithm. In *2019 International Conference on Computer, Information and Telecommunication Systems (CITS)*, pages 1–5. IEEE, 2019.

[35] Dionisios N Sotiropoulos and George A Tsihrintzis. *Machine Learning Paradigms: Artificial Immune Systems and Their Applications in Software Personalization*, volume 118. Springer, 2016.

[36] Dionisios N Sotiropoulos, Aristomenis S Lampropoulos, and George A Tsihrintzis. Artificial immune system-based music genre classification. In *New Directions in Intelligent Interactive Multimedia*, pages 191–200. Springer, 2008.

[37] Dionysios N Sotiropoulos and George A Tsihrintzis. Artificial immune system-based classification in class-imbalanced image classification problems. In *2012 Eighth International Conference on Intelligent Information Hiding and Multimedia Signal Processing*, pages 138–141. IEEE, 2012.

[38] Dionysios N Sotiropoulos and George A Tsihrintzis. Artificial immune system-based classification in class-imbalanced problems. In *2011 IEEE Workshop on Evolving and Adaptive Intelligent Systems (EAIS)*, pages 131–138. IEEE, 2011.

[39] Dionisios N Sotiropoulos and George A Tsihrintzis. Artificial immune system-based classification in extremely imbalanced classification problems. *International Journal on Artificial Intelligence Tools*, 26(03):1750009, 2017.

[40] DN Sotiropoulos, AS Lampropoulos, and GA Tsihrintzis. Artificial immune system-based music piece similarity measures and database organization. In *Proc. 5th EURASIP Conference on Speech and Image Processing, Multimedia Communications and Services. Smolenice, Slovak Republic*, 2005.

[41] Dionisios N Sotiropoulos and George A Tsihrintzis. Artificial immune system-based music recommendation. *Intelligent Decision Technologies*, 12(2):213–229, 2018.

[42] Yoshiteru Ishida. Fully distributed diagnosis by pdp learning algorithm: towards immune network pdp model. In *1990 IJCNN International Joint Conference on Neural Networks*, pages 777–782. IEEE, 1990.

[43] L Nunes De Castro and Fernando J Von Zuben. The clonal selection algorithm with engineering applications. In *Proceedings of GECCO*, volume 2000, pages 36–39, 2000.

[44] Modupe Ayara, Jon Timmis, Rogerio de Lemos, Leandro N de Castro, and Ross Duncan. Negative selection: How to generate detectors. In *Proceedings of the 1st International Conference on Artificial Immune Systems (ICARIS)*, volume 1, pages 89–98. University of Kent at Canterbury Printing Unit University of Kent at Canterbury, 2002.

[45] Julie Greensmith. *The dendritic cell algorithm*. PhD thesis, Citeseer, 2007.

[46] Pete May, Jon Timmis, and Keith Mander. Immune and evolutionary approaches to software mutation testing. In *International Conference on Artificial Immune Systems*, pages 336–347. Springer, 2007.

[47] Vincenzo Cutello, Giuseppe Nicosia, Mario Pavone, and Jonathan Timmis. An immune algorithm for protein structure prediction on lattice models. *IEEE transactions on evolutionary computation*, 11(1):101–117, 2007.

[48] William O Wilson, Phil Birkin, and Uwe Aickelin. Price trackers inspired by immune memory. In *International Conference on Artificial Immune Systems*, pages 362–375. Springer, 2006.

[49] Ma Jie, Gao Hong-yuan, and Diao Ming. Multiuser detection using the clonal selection algorithm and hopfield neural network. In *2006 International Conference on Communications, Circuits and Systems*, volume 2, pages 739–743. IEEE, 2006.

[50] Aihua Wang, Binbin Xu, Ronghua Zhou, and Zhongxia He. Near-optimal mimo multiuser detection using hybrid immune clonal selection algorithm. In *2008 Third International Conference on Communications and Networking in China*, pages 983–986. IEEE, 2008.

[51] Xiao Zhi Gao, Xiaolei Wang, and Seppo J Ovaska. Fusion of clonal selection algorithm and differential evolution method in training cascade–correlation neural network. *Neurocomputing*, 72(10-12):2483–2490, 2009.

[52] Hamed Chitsaz, Nima Amjady, and Hamidreza Zareipour. Wind power forecast using wavelet neural network trained by improved clonal selection algorithm. *Energy conversion and Management*, 89:588–598, 2015.

[53] Leandro N De Castro and Fernando J Von Zuben. Learning and optimization using the clonal selection principle. *IEEE transactions on evolutionary computation*, 6(3):239–251, 2002.

[54] Xin-She Yang and Suash Deb. Cuckoo search via lévy flights. In *2009 World congress on nature & biologically inspired computing (NaBIC)*, pages 210–214. IEEE, 2009.

[55] Amir Hossein Gandomi, Xin-She Yang, and Amir Hossein Alavi. Cuckoo search algorithm: a metaheuristic approach to solve structural optimization problems. *Engineering with computers*, 29(1):17–35, 2013.

[56] Pauline Ong and Zarita Zainuddin. An efficient cuckoo search algorithm for numerical function optimization. In *AIP Conference Proceedings*, volume 1522, pages 1378–1384. American Institute of Physics, 2013.

[57] Momin Jamil and Hans-Jürgen Zepernick. Multimodal function optimisation with cuckoo search algorithm. *International Journal of Bio-Inspired Computation*, 5(2):73–83, 2013.

[58] Fan Wang, XS He, Yan Wang, and SM Yang. Markov model and convergence analysis based on cuckoo search algorithm. *Computer Engineering*, 38(11):180–185, 2012.

[59] Xin-She Yang and Suash Deb. Engineering optimisation by cuckoo search. *International Journal of Mathematical Modelling and Numerical Optimisation*, 1(4):330–343, 2010.

[60] Abdollah Kavousi-Fard and Farzaneh Kavousi-Fard. A new hybrid correction method for short-term load forecasting based on arima, svr and csa. *Journal of Experimental & Theoretical Artificial Intelligence*, 25(4):559–574, 2013.

[61] Milos Madic and Miroslav Radovanovic. Application of cuckoo search algorithm for surface roughness optimization in co2 laser cutting. *Annals of the Faculty of Engineering Hunedoara*, 11(1):39, 2013.

[62] Xiangtao Li and Minghao Yin. A hybrid cuckoo search via lévy flights for the permutation flow shop scheduling problem. *International Journal of Production Research*, 51(16):4732–4754, 2013.

[63] Aziz Ouaarab, Belaïd Ahiod, and Xin-She Yang. Discrete cuckoo search algorithm for the travelling salesman problem. *Neural Computing and Applications*, 24(7-8):1659–1669, 2014.

[64] Ehsan Valian, Saeed Tavakoli, Shahram Mohanna, and Atiyeh Haghi. Improved cuckoo search for reliability optimization problems. *Computers & Industrial Engineering*, 64(1):459–468, 2013.

[65] Aminreza Noghrehabadi, Mohammad Ghalambaz, Mehdi Ghalambaz, and Amir Vosough. A hybrid power series–cuckoo search optimization algorithm to electrostatic deflection of micro fixed-fixed actuators. *International Journal of Multidisciplinary Sciences and Engineering*, 2(4):22–26, 2011.

[66] Koffka Khan and Ashok Sahai. Neural-based cuckoo search of employee health and safety (hs). *International Journal of Intelligent Systems and Applications*, 5(2):76, 2013.

[67] Zhigang Lian, Lihua Lu, and Yangquan Chen. A new cuckoo search. In *International Conference on Intelligence Science*, pages 75–83. Springer, 2017.

[68] Gai-Ge Wang, Amir H Gandomi, Xin-She Yang, and Amir H Alavi. A new hybrid method based on krill herd and cuckoo search for global optimisation tasks. *International Journal of Bio-Inspired Computation*, 8(5):286–299, 2016.

[69] Gai-Ge Wang, Suash Deb, Amir H Gandomi, Zhaojun Zhang, and Amir H Alavi. Chaotic cuckoo search. *Soft Computing*, 20(9):3349–3362, 2016.

[70] S Walton, O Hassan, K Morgan, and MR Brown. Modified cuckoo search: a new gradient free optimisation algorithm. *Chaos, Solitons & Fractals*, 44(9):710–718, 2011.

[71] Abdesslem Layeb. A novel quantum inspired cuckoo search for knapsack problems. *International Journal of bio-inspired Computation*, 3(5):297–305, 2011.

[72] Milan Tuba, Milos Subotic, and Nadezda Stanarevic. Modified cuckoo search algorithm for unconstrained optimization problems. In *Proceedings of the 5th European conference on European computing conference*, pages 263–268. World Scientific and Engineering Academy and Society (WSEAS), 2011.

[73] Manoj Kumar Naik and Rutuparna Panda. A novel adaptive cuckoo search algorithm for intrinsic discriminant analysis based face recognition. *Applied Soft Computing*, 38:661–675, 2016.

[74] Xiangtao Li and Minghao Yin. Modified cuckoo search algorithm with self adaptive parameter method. *Information Sciences*, 298:80–97, 2015.

[75] Xin-She Yang and Suash Deb. Multiobjective cuckoo search for design optimization. *Computers & Operations Research*, 40(6):1616–1624, 2013.

[76] K Chandrasekaran and Sishaj P Simon. Multi-objective scheduling problem: hybrid approach using fuzzy assisted cuckoo search algorithm. *Swarm and Evolutionary Computation*, 5:1–16, 2012.

[77] Leandrodos Santos Coelho, FabioAlessandro Guerra, Nelson Jhoe Batistela, and Jean Vianei Leite. Multiobjective cuckoo search algorithm based on duffing's oscillator applied to jiles-atherton vector hysteresis parameters estimation. *IEEE transactions on magnetics*, 49(5):1745–1748, 2013.

[78] Nandar Lynn, Mostafa Z Ali, and Ponnuthurai Nagaratnam Suganthan. Population topologies for particle swarm optimization and differential evolution. *Swarm and evolutionary computation*, 39:24–35, 2018.

[79] Joshua L Payne, Mario Giacobini, and Jason H Moore. Complex and dynamic population structures: synthesis, open questions, and future directions. *Soft computing*, 17(7):1109–1120, 2013.

[80] Fei Peng, Ke Tang, Guoliang Chen, and Xin Yao. Population-based algorithm portfolios for numerical optimization. *IEEE Transactions on evolutionary computation*, 14(5):782–800, 2010.

[81] Darrell Whitley. A genetic algorithm tutorial. *Statistics and computing*, 4(2):65–85, 1994.

[82] Erick Cantu-Paz. *Efficient and accurate parallel genetic algorithms*, volume 1. Springer Science & Business Media, 2000.

[83] Jing J Liang, A Kai Qin, Ponnuthurai N Suganthan, and S Baskar. Comprehensive learning particle swarm optimizer for global optimization of multimodal functions. *IEEE transactions on evolutionary computation*, 10(3):281–295, 2006.

[84] Bo-Yang Qu, Ponnuthurai Nagaratnam Suganthan, and Swagatam Das. A distance-based locally informed particle swarm model for multimodal optimization. *IEEE Transactions on Evolutionary Computation*, 17(3):387–402, 2012.

[85] Javier Apolloni, José García-Nieto, Enrique Alba, and Guillermo Leguizamón. Empirical evaluation of distributed differential evolution on standard benchmarks. *Applied Mathematics and Computation*, 236:351–366, 2014.

[86] Jingliang Liao, Yiqiao Cai, Tian Wang, Hui Tian, and Yonghong Chen. Cellular direction information based differential evolution for numerical optimization: an empirical study. *Soft Computing*, 20(7):2801–2827, 2016.

[87] Michael Kirley and Robert Stewart. Multiobjective evolutionary algorithms on complex networks. In *International Conference on Evolutionary Multi-Criterion Optimization*, pages 81–95. Springer, 2007.

[88] Andrea Gasparri, Stefano Panzieri, and Federica Pascucci. A spatially structured genetic algorithm for multi-robot localization. *Intelligent Service Robotics*, 2(1):31–40, 2009.

[89] Mario Giacobini, Marco Tomassini, and Andrea Tettamanzi. Takeover time curves in random and small-world structured populations. In *Proceedings of the 7th annual conference on Genetic and evolutionary computation*, pages 1333–1340, 2005.

[90] Chenggong Zhang and Zhang Yi. Scale-free fully informed particle swarm optimization algorithm. *Information Sciences*, 181(20):4550–4568, 2011.

[91] Chuan Wang, Yancheng Liu, Youtao Zhao, and Yang Chen. A hybrid topology scale-free gaussian-dynamic particle swarm optimization algorithm applied to real power loss minimization. *Engineering Applications of Artificial Intelligence*, 32:63–75, 2014.

[92] Emilio Salinas and Laurence F Abbott. A model of multiplicative neural responses in parietal cortex. *Proceedings of the national academy of sciences*, 93(21):11956–11961, 1996.

[93] Fabrizio Gabbiani, Holger G Krapp, Christof Koch, and Gilles Laurent. Multiplicative computation in a visual neuron sensitive to looming. *Nature*, 420(6913):320, 2002.

[94] Robert B Payne and Michael D Sorensen. *The cuckoos*, volume 15. Oxford University Press, 2005.

[95] Andy M Reynolds and Mark A Frye. Free-flight odor tracking in drosophila is consistent with an optimal intermittent scale-free search. *PloS one*, 2(4):e354, 2007.

[96] Clifford T Brown, Larry S Liebovitch, and Rachel Glendon. Lévy flights in dobe ju/'hoansi foraging patterns. *Human Ecology*, 35(1):129–138, 2007.

[97] Ilya Pavlyukevich. Lévy flights, non-local search and simulated annealing. *Journal of Computational Physics*, 226(2):1830–1844, 2007.

[98] Albert-László Barabási and Réka Albert. Emergence of scaling in random networks. *science*, 286(5439):509–512, 1999.

[99] Steven A Hofmeyr and Stephanie Forrest. Architecture for an artificial immune system. *Evolutionary computation*, 8(4):443–473, 2000.

[100] Licheng Jiao and Lei Wang. A novel genetic algorithm based on immunity. *IEEE Transactions on Systems, Man, and Cybernetics-part A: systems and humans*, 30(5):552–561, 2000.

[101] Vincenzo Cutello and Giuseppe Nicosia. The clonal selection principle for in silico and in vitro computing. In *Recent developments in biologically inspired computing*, pages 140–147. Igi Global, 2005.

[102] Arthur Asuncion and David Newman. Uci machine learning repository, 2007.

[103] Ilkay Cinar and Murat Koklu. Classification of rice varieties using artificial intelligence methods. *International Journal of Intelligent Systems and Applications in Engineering*, 7(3):188–194, 2019.

[104] Davide Chicco and Giuseppe Jurman. Machine learning can predict survival of patients with heart failure from serum creatinine and ejection fraction alone. *BMC medical informatics and decision making*, 20(1):16, 2020.

[105] George EP Box, Gwilym M Jenkins, Gregory C Reinsel, and Greta M Ljung. *Time series analysis: forecasting and control*. John Wiley & Sons, 2015.

[106] Salvador García, Daniel Molina, Manuel Lozano, and Francisco Herrera. A study on the use of non-parametric tests for analyzing the evolutionary algorithms' behaviour: a case study on the cec'2005 special session on real parameter optimization. *Journal of Heuristics*, 15(6):617, 2009.

[107] Joaquín Derrac, Salvador García, Daniel Molina, and Francisco Herrera. A practical tutorial on the use of nonparametric statistical tests as a methodology for comparing evolutionary and swarm intelligence algorithms. *Swarm and Evolutionary Computation*, 1(1):3–18, 2011.

[108] George Hripcsak and Adam S Rothschild. Agreement, the f-measure, and reliability in information retrieval. *Journal of the American Medical Informatics Association*, 12(3):296–298, 2005.

[109] Jacob Cohen. A coefficient of agreement for nominal scales. *Educational and psychological measurement*, 20(1):37–46, 1960.

[110] Tom Fawcett. Roc graphs: Notes and practical considerations for researchers. *Machine learning*, 31(1):1–38, 2004.

[111] Joseph Lee Rodgers and W Alan Nicewander. Thirteen ways to look at the correlation coefficient. *The American Statistician*, 42(1):59–66, 1988.

[112] Yu Xue, Jiongming Jiang, Binping Zhao, and Tinghuai Ma. A self-adaptive artificial bee colony algorithm based on global best for global optimization. *Soft Computing*, 22(9):2935–2952, 2018.

[113] Yu Xue, Haokai Zhu, Jiayu Liang, and Adam Słowik. Adaptive crossover operator based multi-objective binary genetic algorithm for feature selection in classification. *Knowledge-Based Systems*, page 107218, 2021.

[114] Jiayu Liang and Yu Xue. An adaptive gp-based memetic algorithm for symbolic regression. *Applied Intelligence*, 50(11):3961–3975, 2020.

[115] Yu Xue, Bing Xue, and Mengjie Zhang. Self-adaptive particle swarm optimization for large-scale feature selection in classification. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 13(5):1–27, 2019.

[116] J. Liang Y. Xue, Y. Wang and A. Slowik. A self-adaptive mutation neural architecture search algorithm based on blocks. *IEEE Computational Intelligence Magazine*, 2021.

[117] Zong Woo Geem and Kwee-Bo Sim. Parameter-setting-free harmony search algorithm. *Applied Mathematics and Computation*, 217(8):3881–3889, 2010.

[118] Young Hwan Choi, Sajjad Eghdami, Thi Thuy Ngo, Sachchida Nand Chaurasia, and Joong Hoon Kim. Comparison of parameter-setting-free and self-adaptive harmony search. In *Harmony Search and Nature Inspired Optimization Algorithms*, pages 105–112. Springer, 2019.

[119] Xingjuan Cai, Xiao-zhi Gao, and Yu Xue. Improved bat algorithm with optimal forage strategy and random disturbance strategy. *International Journal of Bio-Inspired Computation*, 8(4):205–214, 2016.

[120] Rajesh Jugulum, Shin Taguchi, et al. *Computer-based robust engineering: essentials for DFSS*. ASQ Quality Press, 2004.

[121] Annette M Molinaro, Richard Simon, and Ruth M Pfeiffer. Prediction error estimation: a comparison of resampling methods. *Bioinformatics*, 21(15):3301–3307, 2005.

[122] Remco R Bouckaert and Eibe Frank. Evaluating the replicability of significance tests for comparing learning algorithms. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pages 3–12. Springer, 2004.

[123] JJ Liang, BY Qu, PN Suganthan, and Alfredo G Hernández-Díaz. Problem definitions and evaluation criteria for the cec 2013 special session on real-parameter optimization. *Computational Intelligence Laboratory, Zhengzhou University, Zhengzhou, China and Nanyang Technological University, Singapore, Technical Report*, 201212(34):281–295, 2013.

[124] N Awad, M Ali, J Liang, B Qu, and P Suganthan. Problem definitions and evaluation criteria for the cec 2017 special sessionand competition on single ob-

jective real-parameter numerical optimization. nanyang technologial university, jordan university of science and technology and zhengzhou university, singapore and zhenzhou. *Nanyang Technological University, Jordan University of Science and Technology and Zhengzhou University, Singapore and Zhenzhou, China, Tech. Rep*, 201611, 2016.

[125] Frank Wilcoxon. Individual comparisons by ranking methods. In *Breakthroughs in statistics*, pages 196–202. Springer, 1992.

[126] Rainer Storn and Kenneth Price. Differential evolution–a simple and efficient heuristic for global optimization over continuous spaces. *Journal of global optimization*, 11(4):341–359, 1997.

[127] J Michael Johnson and V Rahmat-Samii. Genetic algorithms in engineering electromagnetics. *IEEE Antennas and propagation Magazine*, 39(4):7–21, 1997.

[128] Erik Cuevas, Alonso EchavarríA, Daniel ZaldíVar, and Marco PéRez-Cisneros. A novel evolutionary algorithm inspired by the states of matter for template matching. *Expert Systems with Applications*, 40(16):6359–6373, 2013.

[129] Pranjal Verma, Krishnendu Sanyal, Dipti Srinivsan, and KS Swarup. Information exchange based clustered differential evolution for constrained generation-transmission expansion planning. *Swarm and evolutionary computation*, 44:863–875, 2019.

[130] Xin-She Yang. Flower pollination algorithm for global optimization. In *International conference on unconventional computing and natural computation*, pages 240–249. Springer, 2012.

[131] Junkai Ji, Shuangbao Song, Yajiao Tang, Shangce Gao, Zheng Tang, and Yuki Todo. Approximate logic neuron model trained by states of matter search algorithm. *Knowledge-Based Systems*, 163:120–130, 2019.

[132] Esmat Rashedi, Hossein Nezamabadi-Pour, and Saeid Saryazdi. Gsa: a gravitational search algorithm. *Information sciences*, 179(13):2232–2248, 2009.

[133] Erik Cuevas, Alonso Echavarría, and Marte A Ramírez-Ortegón. An optimization algorithm inspired by the states of matter that improves the balance between exploration and exploitation. *Applied intelligence*, 40(2):256–272, 2014.

[134] Swagatam Das and Ponnuthurai N Suganthan. Problem definitions and evaluation criteria for cec 2011 competition on testing evolutionary algorithms on real world optimization problems. *Jadavpur University, Nanyang Technological University, Kolkata*, pages 341–359, 2010.

[135] Xin-She Yang. Firefly algorithm, levy flights and global optimization. In *Research and development in intelligent systems XXVI*, pages 209–218. Springer, 2010.