Binary Equalization Optimizer Based on Variable Order Transfer Functions to Solve High Dimensional Feature Selection Problem

Jia-Ning Hou, Yu Liu*, Jie-Sheng Wang, Hao-Ming Song, Yu-Cai Wang

Abstract—**Feature selection (FS) is the core concept in the field of both machine learning and data management. FS can eliminate irrelevant or partially related features to improve model performance. Wrapped FS method typically solves the discrete optimization problems by converting continuous values to binary values based on a transfer function. A binary equilibrium optimizer based on variable-order transfer function was proposed to solve high dimensional feature selection problem. Based on the four basic transfer functions of V-shaped, U-shaped, S-shaped and Z-shaped, eight transfer functions with varying order are proposed. Based on the binary equilibrium optimizer, the Relief guided strategy and the KNN classifier, the wrapped feature selection was realized. The simulation experiments were divided into two groups by using 12 medium and high dimensional standard UCI datasets. The first group analyzes the feature selection effect of 8 transfer functions under variable order parameters. In the second group of experiments, each winning parameter in the first group of experiments is selected for comparison. The experiment results prove that a reasonable setting of the order of transfer function can obtain better optimization results, and By analyzing the laboratory data, it can be concluded that the feature selection performance under S1(0.4), S2(1), Z1(2) and V2(3) is better.**

Index Terms—**feature selection, equalization optimizer, variable order transfer function, discrete optimization**

I. Introduction

There are various kinds of data in real life, such as electronic trading data and news report data, which usually have multitude of functions and many of which are irrelevant or redundant. It may have a negative impact on machine learning algorithms, such as increasing model learning time and decreasing model accuracy [1]. FS, as a commonly used data preprocessing method, aims to delete unrelevant and superfluous features to enhance the model generalization manifestation [2]. According to the subset search and evaluation process [3], FS methods can be divided into filter [4], wrapper [5] and embedded [6]. The filter approach does not rely on any learning algorithm, but on the correlation between conditional features and classes. Wrapping method deletes or adds features to feature subsets according to the results of classifier [7]. This process will consume more resources. The optimal feature subset in FS is proved to be a NP-hard problem [8]. Therefore, for high-dimensional data, such as a data set with N features, its search space grows exponentially with the number of features [9], it is difficult to obtain the optimal feature subset. So scholars have designed many search strategies to improve its efficiency [10]. Since meta-heuristic method has strong global searching ability, it can quickly find nearly optimal solution in search space, many studies have proposed meta-heuristic methods (MHO) to solve FS problems [11], such as african vulture optimization algorithm (AVOA) [12], particle swarm optimization (PSO) [13], genetic algorithm (GA) [14], ant colony optimization (ACO) [15], simulating annealing (SA) [16], etc.

In machine learning, discretization is also an important technique for pre-processing data so as to improve the performance of algorithms on high-dimensional data [17]. Because many FS methods require discrete data, it is common practice to discretize the data before performing FS. In addition, in order to improve efficiency, features are usually discrete independently (or with one variable) [18]. The premise of this scheme is based on the assumption that each feature is independent, but when there is interaction between features, this scheme may not be valid. Therefore, univariate discretization may reduce the performance of FS because some information may be lost during discretization due to the interaction between features. Discretization is a common and efficient data structure. It optimizes complex data by establishing a one-to-one mapping relationship between data and storage structure (array) [19]. The most important point of discretization is to establish mappings. For special line segments and points, these mappings can be an interval mapping, that is, mapping a line or an area into an array, so as to reduce the time complexity in calculation [20]. FS is a typical discrete method. Most of the available MHOs are not well suited for straightly obtaining solutions of discrete optimization projects. Effective transfer functions can solve this problem. It also has many applications in

Manuscript received May 17, 2023; revised Aug 1, 2023. This work was supported by the Basic Scientific Research Project of Institution of Higher Learning of Liaoning Province (Grant No. LJKZ0293), and Postgraduate Education Reform Project of Liaoning Province (Grant No. LNYJG2022137).

Jia-Ning Hou is a postgraduate student of School of Electronic and Information Engineering, University of Science and Technology Liaoning, Anshan, 114051, P. R. China (e-mail: 1994905806@qq.com).

Yu Liu is an associate professor of School of Electronic and Information Engineering, University of Science and Technology Liaoning, Anshan, 114051, P. R. China (Corresponding author, phone: 86-0412-2538246; fax: 86-0412-2538244; lnasacl@126.com).

Jie-Sheng Wang is a professor of School of Electronic and Information Engineering, University of Science and Technology Liaoning, Anshan, 114051, P. R. China (e-mail: wang_jiesheng@126.com).

Hao-Ming Song a doctoral student of School of Electronic and Information Engineering, University of Science and Technology Liaoning, Anshan, 114051, P. R. China (e-mail: 823234816@qq.com).

Yu-Cai Wang is a doctoral student of School of Electronic and Information Engineering, University of Science and Technology Liaoning, Anshan, 114051, P. R. China (e-mail: 1275934857@qq.com).

practice. Based on the S-shaped and V-shaped transfer functions, Zhang et al. transform the relief-guided equalizer algorithm into binary to resolving the FS problem [21]. Papa et al. mapped real-valued solutions to Boolean hypercubes with various of transfer functions, and evaluated the binary brainstorming optimization algorithm (BBSO) in different scenarios [22]. Faris et al. converted the continuous salp swarm algorithm (SSA) into binary to solve the related FS problem by use eight transfer functions [23]. Mohamed et al. studied several V-shaped, S-shaped and U-shaped transfer functions, combined the transfer functions with the Gray Wolf optimizer (GWO), converted continuous variables into binary variables so as to solve the high dimension FS problems [24]. Ghosh et al. proposed a new X-type transfer function to enhance the exploration and exploitation capability of binary sequence minimum optimization algorithm (BSMO), which is more effective than other meta-heuristic algorithms in improving the classification accuracy [25]. Ghaith et al. used two transfer functions (Sigmoid function and V-shaped function) to design a binary strategy optimizer (BPO) to screen highly correlated genes from gene expression data and solve the FS problem [26]. Qasim et al. proposes variations of 6 different types of bat algorithm (BA) (BA-s and BA-V), each of which uses a transfer function to map solutions from continuous space to discrete space for implementing FS [27]. With researchers' deeper exploration of transfer functions, the application of time-varying transfer functions has received a certain attention [28]. Researchers transform transfer functions into state spaces (controllable standard or observable standard), reconstruct state space matrix through transfer function coefficients, and realize time-varying transfer functions through state space [29].

FS consists of filter type, wrapper type and embedded type according to its relationship with classification model. Different from filter selection, wrapper feature selection takes the performance of the ultimate learner is used directly as an evaluation criterion for the subset of features. In other words, the aim of the wrapper FS is to select a subset of features for a given learner that is most "tailored" to its performance. The wrapper FS method directly optimizes a given learner. Therefore, from the final performance of the learner, the wrapper FS performs much superior than the filter FS. On the other hand, because the learner needs to be trained many times in the FS process, the calculation cost of wrapper FS is usually much higher than that of filter FS. Generally, the wrapper FS method has better classification performance than the filter FS method, and the feature subset obtained is often more in line with the requirements of the classifier, and has been widely used in practice. Alagu et al. uses an improved wrapper FS technique and combines KNN and SVM for classification [30]. Hoou et al. proposed an improved wrapper FS technique and integrated it with a support vector machine (SVM) classifier as a component of a complete fault diagnostic system for the case research on rolling bearings [31]. Tran et al. uses wrapper FS to improve classification performance of incomplete data [32]. Zhang et al, also used KNN classifier when they studied the feature selection method based on henry gas solubility algorithm [33]. Wrapper based feature subset selection (FSS) methods typically obtain higher classification accuracies than filtering methods, but they require additional time, particularly for applications with several thousand functions, such as analysis of microarray data [34]. Chang et al. proposed a multi-layer feed-forward neural network (MFNN) model based on the wrapper approach as the best prediction for inferring disease susceptibility based on the sophisticated relationship between osteoporosis and single nucleotide polymorphism (SNP) in Taiwanese women [35]. Alikhani et al. proposed a wrapper FS method for accurate detection of rotating diode rectifier faults in brushless synchronous generators [36].

Equilibrium optimizer (EO) is a fresh intelligent algorithm presented by Faramarzi in 2019. It simulates the process of dynamic mass balance in volume and sets a concentration balancing pool. The process of dynamic equilibrium of each concentration towards the equilibrium pool is the process of population convergence to the global optimal solution [37]. Contrast that with other swarm intelligence algorithms, such as genetic algorithm (GA) and whale optimization algorithm (WOA), it utilizes various mechanisms for position update and has the characteristics of simple structure and high flexibility, and has been further applied by scholars to scheduling optimization and other engineering problems [38-40]. In this paper, a binary equalization optimizer based on variable-order transfer function is proposed for high-dimensional feature selection. Eight transfer functions (three V-shaped, one U-shaped, two S-shaped and two Z-shaped) are selected and the order of each transfer function is set. Based on the binary equalization optimizer (BEO), KNN was used as the classifier and the Relief guidance strategy was used to conduct two sets of simulation experiments for each transfer function on 12 medium and high dimensional UCI datasets.

II. Binary Equalization Optimizers

A. Equalization Optimizer

EO is a new intelligent algorithm that shows good advantages in many test functions compared with the traditional intelligent algorithms. The equilibrium equations for mass describe the processes by which mass entered, left, and was generated in a control volume, which can be expressed by the first differential equation.

$$V\frac{dC}{dt} = QC_{eq} - QC + G \tag{1}$$

where, V is the control volume, C is the concentration in the control volume, Q is the volume rate of flow come or out of the control volume, C_{eq} is the concentration when the control volume is in equilibrium, G is the mass generation rate in the control volume, and $V(dC/dt)$ is the mass change rate of the control volume. When it exhibits 0, it indicates that the control volume is in steady state equilibrium. Set $\lambda = Q/V$ and transform Eq. (1) as:

$$\frac{\mathrm{d}C}{\lambda C_{\mathrm{eq}} - \lambda C + \frac{G}{V}} = \mathrm{dt} \tag{2}$$

Let t_0 and C_0 are the original time and concentration values, respectively. By simultaneously integrating both sides of Eq. (2), obtain:

$$C = C_{eq} + (C_0 - C_{eq})F + \frac{G}{\lambda V}(1 - F) \tag{3}$$

where, $F = \exp(-\lambda(t - t_0))$.

Algorithm model is described as follow.

(1) initialization. Similar to most MHO, the initialization process of the EO can be showed as:

$$C_i^{\text{init}} = C_{\min} + \text{rand}_i (C_{\max} - C_{\min}), i = 1, 2, ..., n \tag{4}$$

where, C_i^{init} is the initial concentration vector of the i-th individual, $C_{\min}$ and $C_{\max}$ are the lower and upper limit vectors of individuals, rand_i is a random vector in the scope of $[0,1]$, and n is the number of individuals in the population.

(2) Establish a balance pool. The equilibrium state is the final state where the algorithm converges. In the optimization process, the function of the equilibrium pool is to provide candidate solutions for the whole optimization process. The four individuals with the most fitness values generated by the current initialization iteration process and their average values form the balance pool.

$$\overrightarrow{C_{eq.pool}} = \left\{\overrightarrow{C_{eq(1)}}, \overrightarrow{C_{eq(2)}}, \overrightarrow{C_{eq(3)}}, \overrightarrow{C_{eq(4)}}, \overrightarrow{C_{eq(ave)}}\right\} \tag{5}$$

where, $\overrightarrow{C_{eq(ave)}} = (\overrightarrow{C_{eq(1)}} + \overrightarrow{C_{eq(2)}} + \overrightarrow{C_{eq(3)}} = \overrightarrow{C_{eq(4)}})/4$. The five individuals in the equilibrium pool that are likely to be selected as the solution to the optimization process have the same probability, that is, all of them are 0.2.

(3) Exponential term. Exponential term is an important part of algorithm update, which can be expressed as:

$$\vec{F} = e^{-\vec{\lambda}(t - t_0)} \tag{6}$$

where, $\vec{\lambda}$ is a random vector between 0 and a, and the variable t is defined as a function that declines with each iterations, namely:

$$t = \left(1 - \frac{Iter}{Max\_iter}\right)^{a2\frac{Iter}{Max\_iter}} \tag{7}$$

where, $Iter$ and Max_iter are the current iteration times and the maximum iteration times, and a_2 is a constant, generally taking 1.

To ensure the convergence of the algorithm and improve the search and utilization ability of the algorithm, consider:

$$\vec{t_0} = \frac{1}{\vec{\lambda}} \ln\left(-a_1 sign\left(\vec{r} - 0.5\right)\left[1 - e^{-\vec{\lambda}t}\right]\right) + t \tag{8}$$

=where, a_1 is a constant, generally set as 2 and $\vec{r}$ is a random vector between $[0,1]$. Substitute Eq. (8) into Eq. (6) to get:

$$\vec{F} = a_1 sign\left(\vec{r} - 0.5\right)\left[e^{-\vec{\lambda}t} - 1\right] \tag{9}$$

(4) Generation rate.

$$\vec{G} = \vec{G_0} e^{\vec{k}(t - t_0)} \tag{10}$$

For a more manageable and more systematic searching pattern, $k = \lambda$. The exponential term derived before is introduced, and the generation rate is described as:

$$\vec{G} = \vec{G_0} e^{\vec{\lambda}(t - t_0)} = \vec{G_0}\vec{F} \tag{11}$$

$$\overrightarrow{G_0} = \overrightarrow{GCP}\left(\overrightarrow{C_{eq}} - \vec{\lambda}\vec{C}\right), \overrightarrow{GCP} = \begin{cases} 0.5r1, r_2 \geq GP \\ 0, r_2 < GP \end{cases} \tag{12}$$

where, r_1 and r_2 are the random numbers between $[0,1]$, and GP is the generatrix probability, which is generally 0.5. To sum up, the final update formula of the balance optimizer is described as:

$$\vec{C} = \overrightarrow{C_{eq}} + \left(\vec{C} - \overrightarrow{C_{eq}}\right)\vec{F} + \frac{\vec{G}}{\vec{\lambda}V}\left(1 - \vec{F}\right) \tag{13}$$

where, the value of V is generally equal to 1.

B. Binary Equalization Optimizer

Binary equalizer optimizer algorithm is obtained by using transfer function based on the currently obtained optimal value and position vector. Through the above two ways, the continuous search space of the basic EO algorithm is converted into discrete binary search space, which can broaden the scope of EO so as to be widely used to solve discrete problems. Fig. 1 is the flowchart of the binary equalization optimizer (BEO). It can be seen from Fig. 1 that the first step is to initialize the population, and then determine whether the number of iterations is less than the maximum number of iterations. If not, return to the best value and end the algorithm; If yes, go on to the next step, calculate the fitness function value of each particle and update the parameters and concentration. Then update the elements by FS calculation probability in the next iteration and compare the results with the transfer function to be compared. Select the required probability value and then the number of iterations add one for the next probability update.

III. BEO Based on Variable Order Transfer Functions to Solve High-dimensional Feature Selection Problem

A. Transfer Function

The first step in the design is to initialize the population. That is, each particle generates a random binary vector $X_i(i = 1,2,...N)$ in the population , where N is the size of the population. For each d-th dimension of the i-th particle, the probability of a binary value 0 or 1 is 0.5. If it is greater than 0.5, it is 1, otherwise it is 0. The use of transfer functions is one of the most efficient ways to convert a continuous algorithm to a binary algorithms for the reasons explained in the above narrative. Next, eight transfer functions are used to transform sequential EO to binary. Table 1 shows the expressions of eight transfer functions, and Fig. 2 shows the schematic diagram of eight transfer functions. Eq. (14)-(15) are two typical V-shaped transfer functions, and Eq. (16) is one typical U-shaped transfer function.

$$T_1\left(X_i^d\right) = \left|\left(\frac{e^{10X_i^d} - 1}{e^{10X_i^d} + 1}\right)^m\right| \tag{14}$$

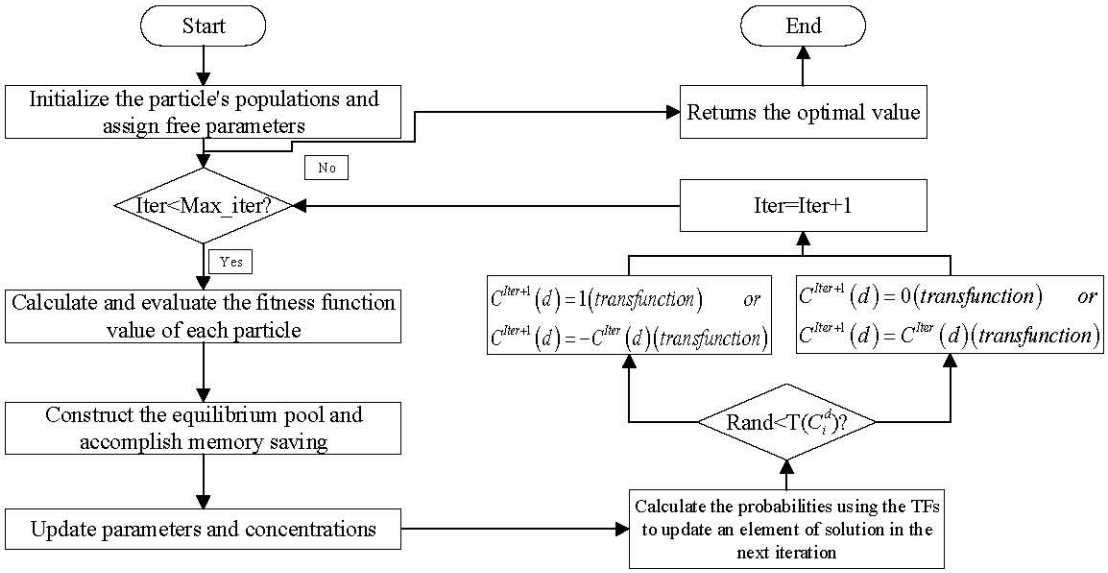


Fig. 1 Flowchart of binary equalization optimizer.

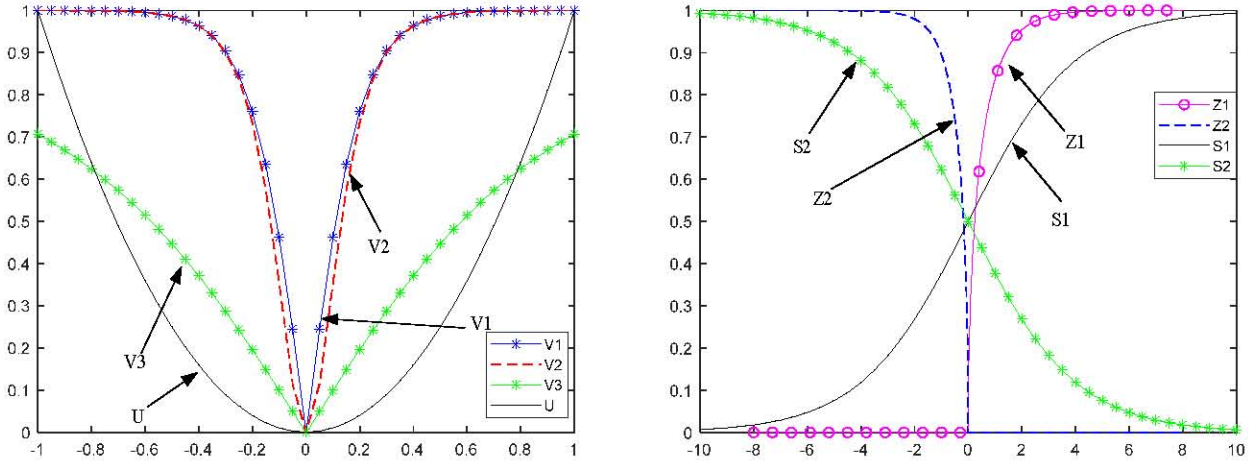


Fig. 2 Eight kinds of transfer functions.

TABLE 1. EXPRESSIONS OF EIGHT TRANSFER FUNCTIONS

Serial number	Name	Expression	Serial number	Name	Expression
1	V1 (m)	$\left\lvert\left(\frac{e^{10x}-1}{e^{10x}+1}\right)^m\right\rvert$	5	S1 (m)	$\frac{1}{1+e^{-mx}}$ (m>0)
2	V2 (m)	$1-\mathrm{sech}^m(10x)$	6	S2 (m)	$\frac{1}{1+e^{-mx}}$ (m<0)
3	V3 (m)	$\left\lvert\left(\frac{x}{\sqrt{1+x^2}}\right)^m\right\rvert$	7	Z1 (m)	$\sqrt{1-m^x}$ (m>1)
4	U (m)	$\lvert x^m\rvert$	8	Z2 (m)	$\sqrt{1-m^x}$ (0<m<1)

$$T_3\left(X_i^d\right)=\left\lvert\left(\frac{X_i^d}{\sqrt{1+(X_i^d)^2}}\right)^m\right\rvert \tag{15}$$

$$T_4\left(X_i^d\right)=\left\lvert(X_i^d)^m\right\rvert \tag{16}$$

where, X_i^d denotes the position of the *i*-th particle in the *d*-th dimension in the EO, and X_i is calculated by Eq. (13). The actual particle concentration is renewed by the probability values gotten in Eq. (14)-(16).

$$X_{t+1}^d=\begin{cases}-X_t^d, If\ rand<T\left(X_t^d\right)\\ X_t^d,\ If\ rand\geq T\left(X_t^d\right)\end{cases} \tag{17}$$

With Eq. (17), we can find that if the random number is less than the probability in Eqs. (14)-(16), the *d*-th dimension of the original individual is negative; Otherwise, it remains the same. Eq. (18)-(20) are the typical V-type, S-type and Z-type transfer functions, respectively, where X_i^d is signified as above and X_i is calculated from Eq. (18)-(20).

$$T_2\left(X_i^d\right)=1-\mathrm{sech}^m(10X_i^d) \tag{18}$$

$$T_5\left(X_i^d\right)=\frac{1}{1+e^{-mX_i^d}} \tag{19}$$

$$T_6\left(X_i^d\right)=\sqrt{1-m^{X_i^d}} \tag{20}$$

The information about the current particle concentration is updated based on the probability values derived from Eqs. (18)-(20).

$$X_{t+1}^{d}=\begin{cases}1, If\ rand<T\left(X_{t}^{d}\right)\\0, If\ rand\geq T\left(X_{t}^{d}\right)\end{cases} \tag{21}$$

where, X_{t+1}^{d} denotes the d-th dimensional factor of particle concentration in the next iteration, and $rand$ is a random number in the range of [0,1]. Through Eq. (18) and Eq. (21), the continuous variables are successfully mapped to binary variables.

B. Variable Order Transfer Functions

Next, the parameter order of eight transfer functions is introduced. Five orders are set for V1, V2, V3 and U according to the upper and lower limits of each function and the parameter setting of X, etc., and four orders are set for S-shaped transfer functions and Z-shaped transfer functions. Fig. 3 shows the transfer functions with variable order and its schematic diagram. Then, based on BEO algorithm, the eight transfer functions are simulated and tested on 12 medium and high dimensional data sets, and the order most suitable for each transfer function is obtained.

C. K-Nearest Neighbor Algorithm

K-Nearest Neighbor (KNN) is one of the simplest algorithms in data mining classification technology. The principle of implementation of KNN classification algorithm is abbreviated as follows. In order to identify the class of the anonymous sample, the distance between the anonymous sample and all already known samples is calculated by using all known samples as reference, from which the K known samples with the closest distant from the anonymous sample are selected. According to the minority-majority voting rule, the unknown sample and the K nearest samples are categorized into one class.

D. Fitness Function

The fitness function of an individual is non-negative, and the larger the fitness value, the better the individual is. The objective function has positive and negative, and there are many kinds of relations between them. For example, when the objective function is the smallest, the larger the adaptive value is, and when the objective value is the largest, the larger the adaptive value is. The key point that affects the degree of convergence of the algorithm and the ability to find an optimal solution is the choice of the fitness function, since the heuristic is directly based on the fitness function during the search. Since the complexity of the fitness function is a major component of the heuristic complexity, the design of the fitness function should be as simple as possible with minimal computational time complexity. The fitness function for feature selection is designed as follows:

$$fitness=h_1\gamma_R(D)+h_2\frac{|M|}{|N|} \tag{22}$$

where, $\gamma_R(D)$ represents the classification error rate corresponding to the feature subset currently selected by the classifier, $|M|$ represents the number of currently selected features, $|N|$ is the total number of features, and h_1 and h_2 are two weight coefficients reflecting the classification rate and length of the subset, satisfying $h_1+h_2=1$, and h_1 and h_2 are set to 0.99 and 0.01, respectively.

IV. Experimental Simulation and Results Analysis

In this section, the proposed BEO with variable-order transfer function is tested to solve the high-dimensional feature selection problem, and the experimental results are analyzed. The performance of the proposed model will be tested on 12 datasets selected from UCI machine learning databases, and two groups of control experiments are set up. Firstly, the BEO algorithms based on variable-order transfer functions are compared, and the each transfer function with optimal parameters is selected for comparison again. Subsection 4.1 describes the experimental parameters and subsection 4.2 describes the datasets for the experiments. Subsection 4.3 gives the performance criteria used to evaluate the models. Finally, Subsection 4.4 presents the analysis of the results and their comparison with other feature selection algorithms.

A. Description of Classification Datasets

Twelve high-dimensional datasets are selected from UCI for classification research. Table 2 lists the details of these high-dimensional datasets.

B. Experimental Parameter Settings

In this study, K-nearest neighbor (KNN) algorithm with Euclidean distance and K=5 is used to calculate the classification accuracy as fitness function. To prevent over-fitting, the data set is divided into training and testing samples as follows. In the first iteration, 80% of the feature vectors are used for to train and the remaining 20% are used for to test. Next, another 20% of the feature vectors are used for testing and the remaining 80% are used for the training. This process is repeated until all the feature vectors are used to test the proposed algorithm. Finally, the average measurement of statistics is collected over 30 separate runs and displayed as the resulting results.

The population size of the algorithm is set to 10, the maximum number of iterations is 100, and the dimension of search space is equal to the total number of features. In the EO algorithm, set parameters $a1$=2, $a2$=1 and GP=0.5.

C. Feature Selection Performance Evaluation Index

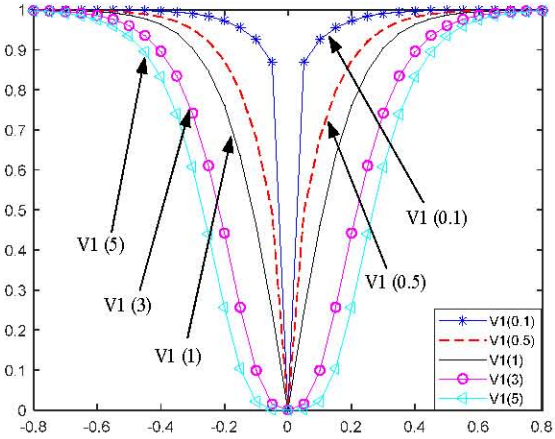
When the results of feature selection are evaluated and interpreted, measurement standards are often used. Here, the evaluation indexes such as average value, standard deviation, average calculation time and average number of features are used. The classification accuracy is calculated by the following standard.

$$Accuracy=\frac{1}{N}\sum_{i=1}^{N}match(Pl_i,Al_i) \tag{23}$$

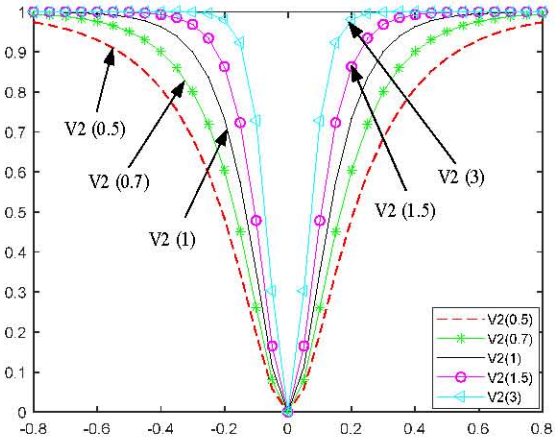
where, N represents the number of testing set points, Pl_i is the predicted data point, Al_i is the actual class in the labeled data, and $match(Pl_i,Al_i)$ is a comparison discriminant function. When $Pl_i=Al_i$, $match(Pl_i,Al_i)=1$, otherwise $match(Pl_i,Al_i)=0$.

$$fitness=0.99*(1-Accuracy)+0.01*\frac{|SelectedfeaturesCount|}{|TotalfeaturesCount|} \tag{24}$$

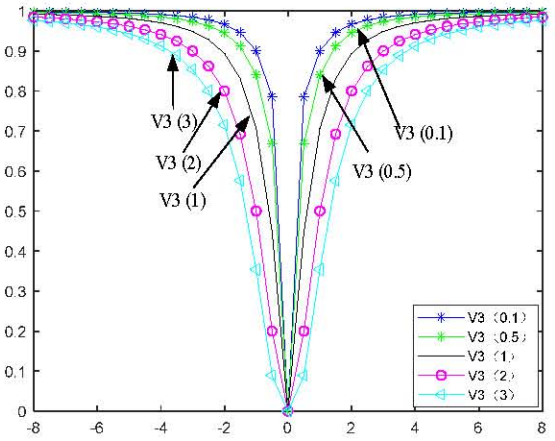
where, $Accuracy$ is obtained by Eq. (23).



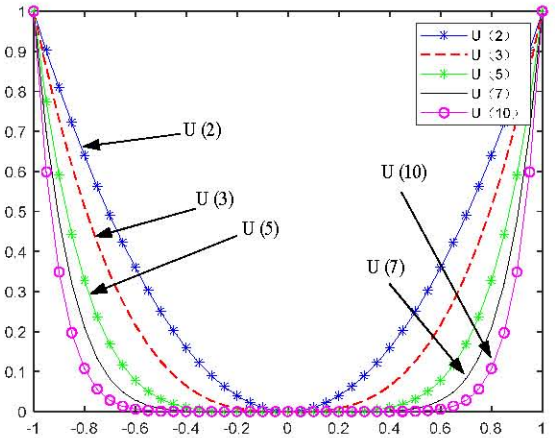
(a) V1



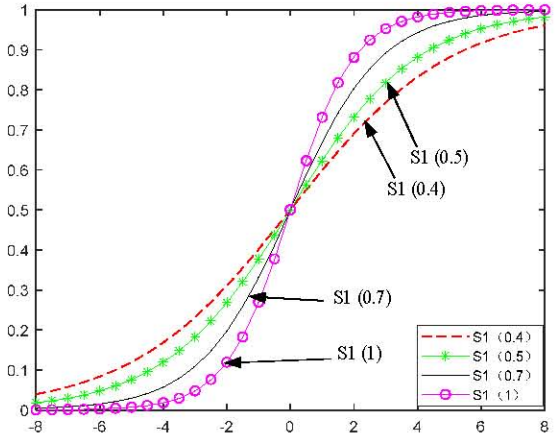
(b) V2



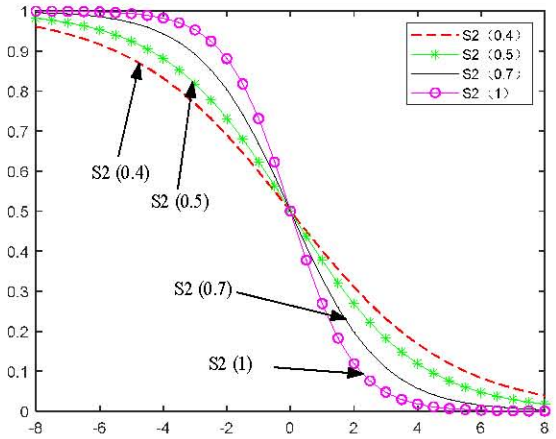
(c) V3



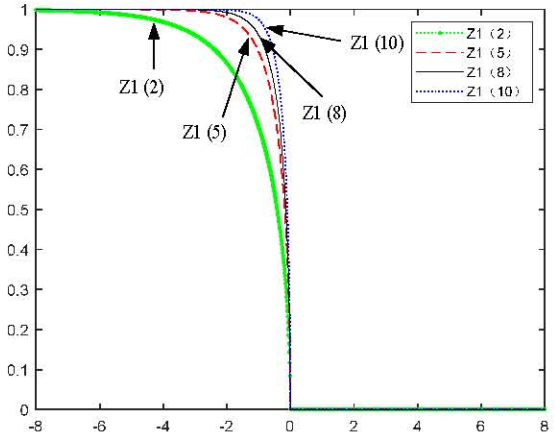
(d) U



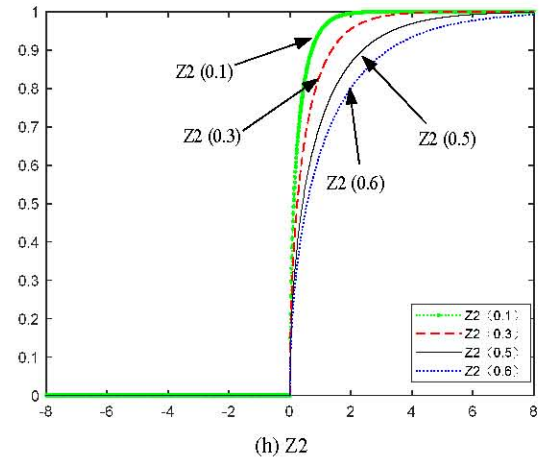
(e) S1



(f) S2



(g) Z1



(h) Z2

Fig. 3 Schematic diagram of variable order transfer functions.

TABLE 2. INFORMATION OF DATASETS

Datasets	Features	Instances	Classes	Datasets	Features	Instances	Classes
Arrhythmia	279	452	3	COIL20	1024	1440	20
Brain Tumor1	5920	90	5	DataMatrix	1024	165	15
Brain Tumor2	10367	50	3	DLBCL	5469	77	2
Breast3	4869	95	2	DrivFace	6400	606	3
Yale	1024	165	15	Lung discrete	325	73	9
Cancer	12600	203	4	Lymphoma	4026	96	9

D. Comparison and Discussion of Simulation Results

In order to verify the order varying performance of the eight transfer functions, two sets of controlled experiments were carried out and the comparison of the performance of the eight different order transfer functions was analyzed. In the first group, 12 groups of experiments were simulated based on the EO in order to verify the validity of the proposed transfer function order. These eight groups of transfer functions are V1, V2 and V3 of the V-type transfer functions, and U-type, S-type and Z-type transfer functions. The second group of experiments is to select the transfer function with the best order in the first group, and continue the simulation experiment on the EO algorithm to select the transfer function and parameters with the best effect. EO algorithm adopts binary version.

(1) Solve Feature Selection Problems Based on Eight Transfer Functions with Different Parameters

1) V1-shaped transfer functions

Five parameters of 0.1, 0.5, 1,3 and 5 are set in the V1-shaped transfer function, and the simulation results are shown in Fig. 4. Table 3 shows the average adaptation and standard deviation of the algorithm, and Tables 4 and Table 5 show the average running time and the number of selected features, respectively. In these tables, the best results are highlighted in bold. In Table 3, V1(5) got the best average adaptation value and standard deviation of adaptation, the average adaptation value got the lowest value in 12 data sets, and the standard deviation of adaptation got the lowest value in 9 data sets, which wins with absolute advantage. The second is V1(0.5), which achieved the best standard deviation of fitness on three data sets. V1(0.1), V1(1) and V1(3) have the lowest effect. In Table 4, there is no doubt that V1(5) has the best effect, and it also wins with absolute advantage in 12 data sets. At the same time, it can be seen from the data that the effect is better with the increase of the order, and the data values are not much different, so it can be concluded that the higher the order, the better the effect. In Table 5, V1(5) is the best in six data sets, followed by V1(3) and V1(0.1), which are the best in four data sets and two data sets respectively, and V1(0.5) and V1(1) were 0.

2) V2-shaped transfer functions

Five parameters, 0.5, 0.7, 1, 1.5 and 3, are set in V2-shaped transfer function. The simulation results are shown in Fig. Table 6 shows the average fitness and standard deviation of the algorithm, and Tables 7 and Table 8 show the average running time and the number of selected features respectively. In these tables, the best results are highlighted in bold. In Table 6, V2(3) got the best average fitness value and standard deviation of fitness, the average fitness value got the lowest value in four data sets, and the standard deviation of fitness got the lowest value in six data sets. The second is V2(0.5), which achieved the best standard deviation of fitness on six data sets.

V2(0.7), V2(1) and V2(1.5) have little difference, and the effect is average. In Table 7, V2(3) has the best effect, which also wins with absolute advantage in 10 data sets, followed by V2(1), which is the best in two data sets. In Table 8, V2(3) is the best in eight data sets, followed by V2(1) and V2(1.5), all of which are the best in one data set, and V2(0.5) and V2(0.7) were 0. It is obvious from Fig. 5(j) that V2(0.5) and V2(0.7) converge most rapidly.

3) V3-shaped transfer functions

Five parameters, 0.3, 0.5, 1, 2 and 3, are set in the V3 transfer function. The simulation results are shown in Fig. 6. Table 9 shows the average fitness and standard deviation of the algorithm. Table 10 and Table 11 show the average running time and the number of selected features, respectively. The best results are marked in bold. In Table 9, the average fitness value of V3(0.3) is the best in 10 datasets and standard deviation is the best in 6 datasets i.e. V3(0.3) is the best. The next best is V3(3) which has the best mean fitness in two data sets and its standard deviation of fitness is the best in five data sets, thus it is the second best. The rest of V3(0.5), V3(1) and V3(2) are approximately the same. In Table 10, V3(2) has the best performance with the lowest values in the seven datasets. It is followed by V3(3) with the best performance on five datasets and the other three parameters are 0. In Table 11, V3(3) wins by a wide margin and achieves the best performance on 11 datasets. As can be seen from the convergence plot of the transfer function in Fig. 6, the majority of the data sets V3(0.3) and V3(0.5) are at the top, and some of them even show convergence values at the beginning, which is extremely unstable and poor performance. In summary, V3(2) is the best of the five parameters.

4) U-shaped transfer functions

Five parameters, 2, 3, 5, 7 and 10, are set in the U-shaped transfer function. The simulation results are shown in Fig. 7. Table 12 shows the average fitness and standard deviation of the algorithm, and Tables 13-14 show

the average running time and the number of selected features respectively. The best results in these tables are highlight in bold. In Table 12, the standard deviations of fitness of U(2) and U(10) are the best in six data sets, U(5) and U(7) are the best in three data sets, and U(3) is zero. From the average value of fitness, U (2) is the best on all 10 data sets, and U (3) is the best on 3 data sets. According to the average time in Table 13, U(10) got the fastest result in nine data sets, followed by U(7) and U(5) getting the best value in two or one data sets, respectively. According to the average of the selected features in Table 14, U(10) is also the best in 9 data sets, followed by U(7) in 3 data sets. Therefore, U(10) can be selected if the convergence time is fast, the number of selected features is large, and the average fitness is not particularly good. As can be seen from Fig. 7, the convergence curves of five parameters are similar in most data sets, but U(2) is always above other parameters and unstable. To sum up, the best parameter is U(10).

5) S-shaped transfer functions

S-shaped transfer functions are divided into S1-shaped and S2-shaped transfer functions. Four parameters of 0.4, 0.5, 0.7 and 1 are set for S1-shaped and S2-shaped transfer functions. The simulation results are shown in the following Fig. 8 and Fig. 9. Tables 15 and Table 18 show the average and standard deviation of the fitness of two algorithms. Tables 16-17 and Table 19-20 show the average of different transfer functions running times and the number of selected features, respectively. In S1 transfer function, seen from Table 15, the fitness average under S1(0.4) is the best in five data sets, S1(0.5) is the best in four data sets, S1(0.7) is the best in five data sets and S1(1) is the best in ten data sets.

For adaptation standard deviation, S1(0.5) and S1(1) achieved best results on 10 and 9 datasets respectively, while the remaining S1(0.4) and S1(0.7) achieved best results on 6 and 5 datasets respectively. From Table 16-17, S1(0.4) is the best in terms of time and number of features, with 8 data sets and 7 data sets respectively, while S1(0.5) is the second best, with 4 and 3 data sets respectively. From the convergence curves of Fig. 8, the S1(1) curve is extremely unstable, sometimes above and sometimes below other parameter curves. It can be seen from Fig. 8.(i)-(j) that the overall parameters converge quickly. To sum up, the parameter 0.4 is the best for S1. Among the S2 transfer functions, seen from Table 18, the fitness average under S2(0.4) is the best in six data sets, S2(0.5) is the best in nine data sets, S2(0.7) is the best in six data sets, and S2(1) is the best in five data sets. For the standard deviation of fitness under S2(0.5) and S2(0.7), seven data sets are the best, and the remaining S2(0.4) and S2(1) are the best. Seen from Table 16-17, S2(1) is the best in terms of time and number of features, with 6 and 11 data sets, and S2(0.7) is the best in average time with 5 data sets. To sum up, the parameter 1 is the best for S2.

6) Z-shaped transfer functions

The Z-shaped transfer functions are divided into Z1-shaped and Z2-shaped transfer functions. Four parameters, 2, 5, 8, 10 and 0.1, 0.3, 0.5 and 0.6, are set for Z1 and Z2 respectively. The simulation results are shown in Fig. 10 and Fig. 11. Table 21 and Table 24 show the average fitness and standard deviation of two algorithms. Table 22-23 and Table 25-26 show the average running time of different transfer functions and the number of selected features respectively.

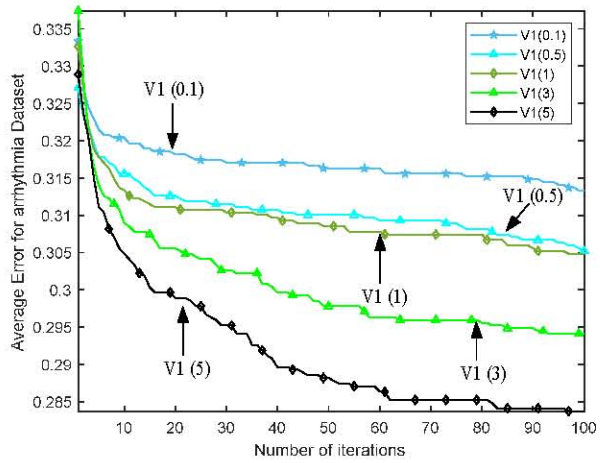
Among the Z1 transfer functions, seen from Table 21, the fitness average under Z1(2) is the best in 12 data sets by absolute advantage. As for the standard deviation of fitness , The best performance was achieved on 10 datasets under Z1(10), both Z1(2) and Z1(5) on 4 datasets, and the remaining Z1(8) on 3 datasets. According to Table 22-23, Z1(2) achieved the best results in 9 data sets in terms of time, and 4 data sets achieved the best results in terms of the number of features. However, in terms of the number of features selected in Table 23, all parameters have optimal values, and the distribution is relatively average. To sum up, the parameter 2 is the best for Z1-shaped transfer functions. Among the Z2 transfer functions, seen from Table 24, the fitness average and standard deviation Z2(0.6) are the best in 10 data sets and 12 data sets, and the fitness standard deviations of Z2(0.1), Z2(0.5) and Z2(0.3) are the best in 5, 4 and 3 data sets respectively. Seen from Table 25-26, Z2(0.6) is the best in terms of time and number of features, with 12 and 11 data sets respectively. It can be clearly seen from Fig. 11 that the curve of Z2(0.1) is unstable. To sum up, the parameter 0.6 is the best for Z2-shaped transfer functions.

(2) Solve Feature Selection Problems Based on Eight Transfer Functions with Optimal Parameters

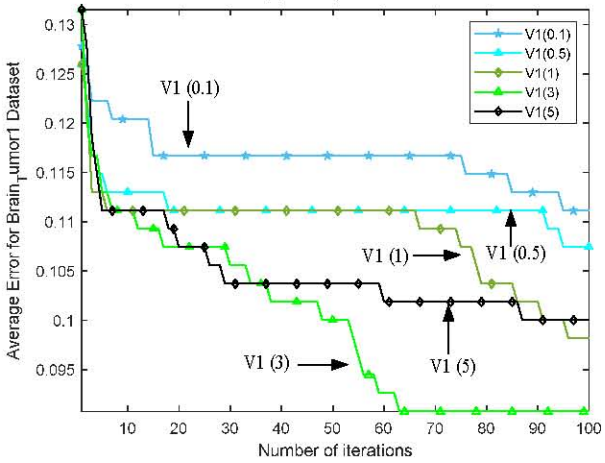
In this section, the optimal parameters of eight transfer functions are selected for carring out simulation experiments, and the 12 middle and high-dimensional data sets are selected. The results are evaluated based on the mean and standard deviation of the fitness, the number of selected features and running time. Fig. 12 shows the convergence curves for each of the eight algorithms, and Table 27-29 shows the statistical data results. Through the analysis of the data in these tables, from the average value of fitness, V2(3) has the best performance, with the lowest value in all seven data sets. Followed by S1(0.4), the lowest values are in five data sets. Followed by V1(5) and Z1(2), all are four data sets. From the standard deviation of fitness, S1(0.4) and S2(1) are the best, with the minimum values of 9 and 8 data sets, respectively, and the other six transfer functions have little difference. It can be seen from Table 28 that Z1(2) has the fastest running time, with absolute advantage, and 12 data sets are the best. In Table 29, Z1(2) and V3(2) are the best in the number of selected features. Through the analysis of the above data, S1(0.4), S2(1), Z1(2) and V2(3) perform better.

V. Conclusion

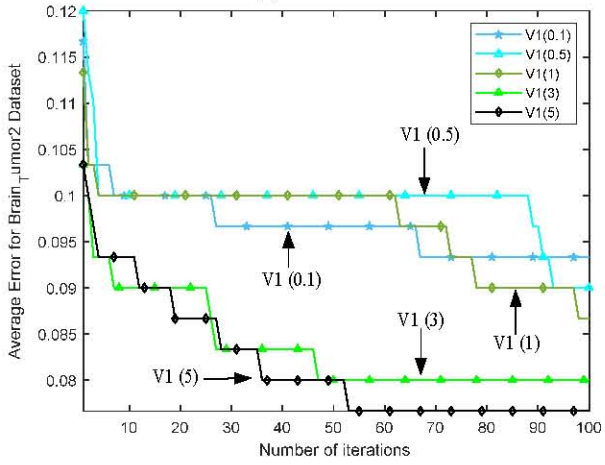
By analyzing the order parameters of eight transfer functions, the transfer function converts the continuous search space in EO to a binary search space and establishes a good balance between exploration and utilization of the binary search space, which is significant for enhancing the performance of FS. Through the experimental comparison on 12 UCI datasets, through the analysis of classification accuracy and feature reduction rate, two groups of simulation experiments are carried out.



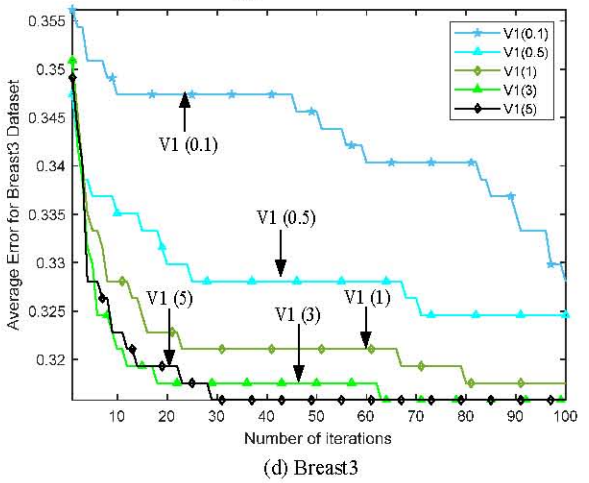
(a) Arrhythmia



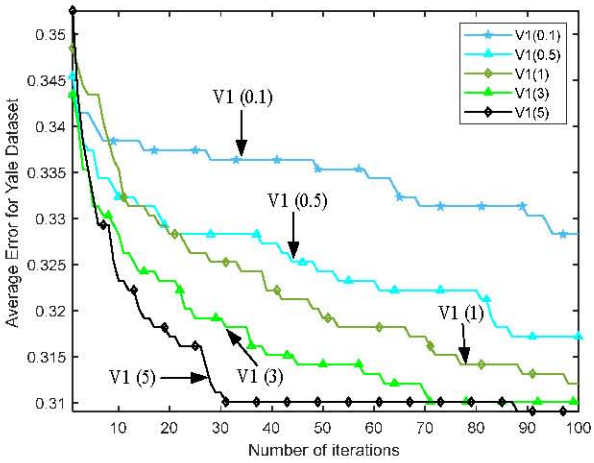
(b) Brain Tumor1



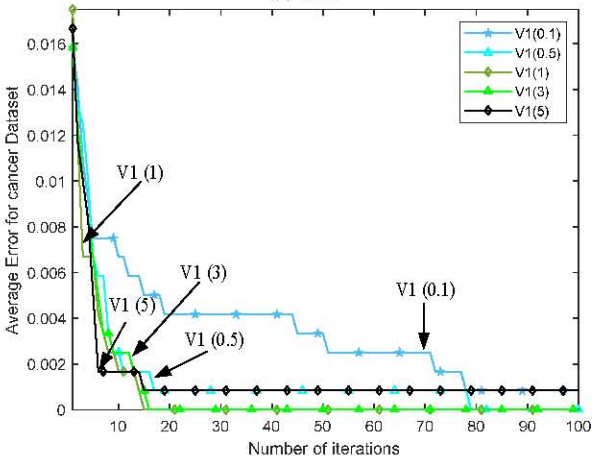
(c) Brain Tumor2



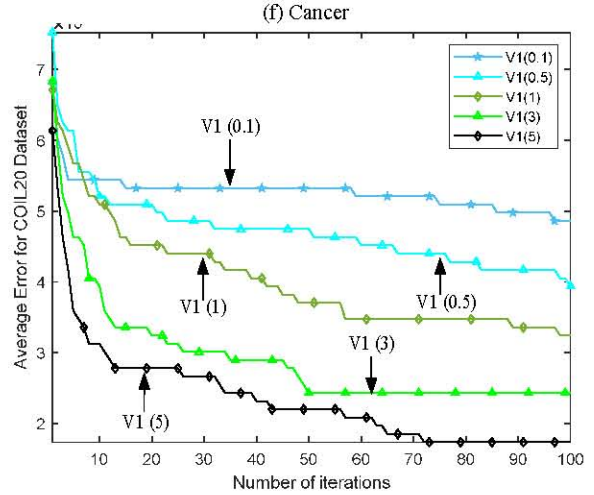
(d) Breast3



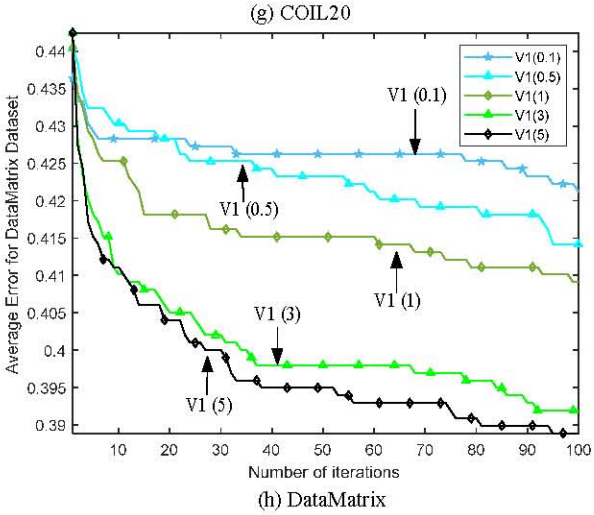
(e) Yale



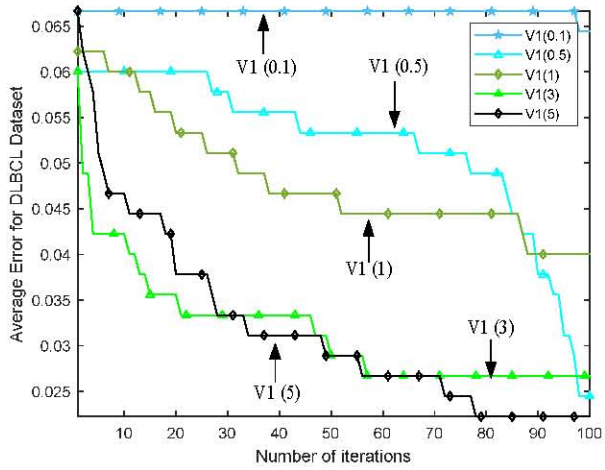
(f) Cancer



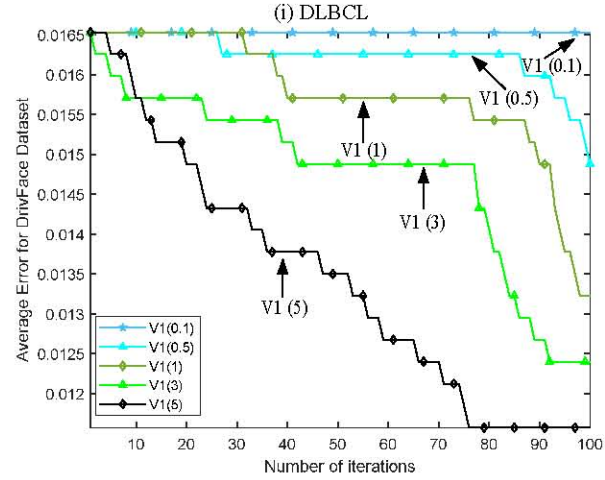
(g) COIL20



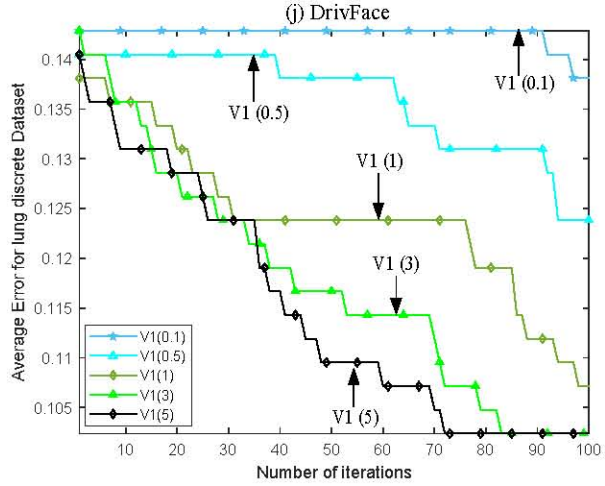
(h) DataMatrix



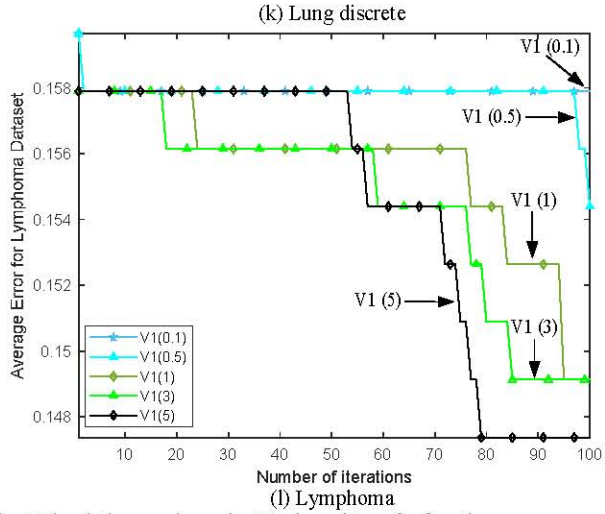
(i) DLBCL



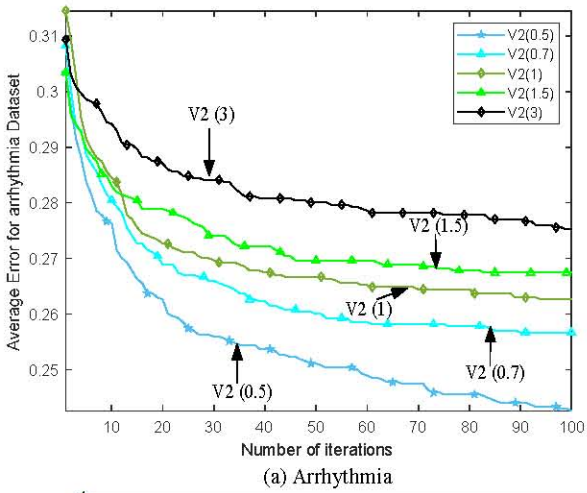
(j) DrivFace



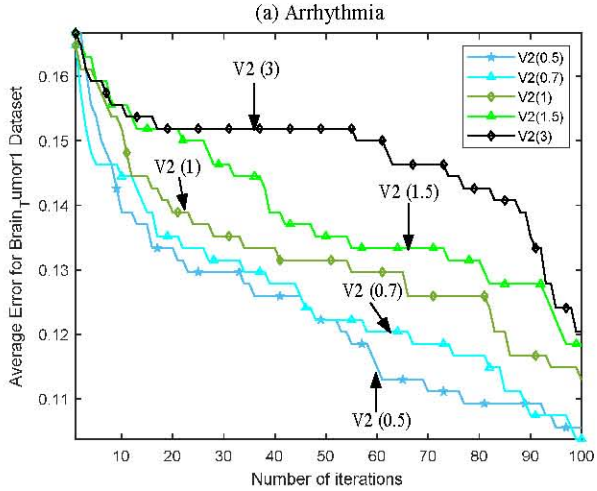
(k) Lung discrete



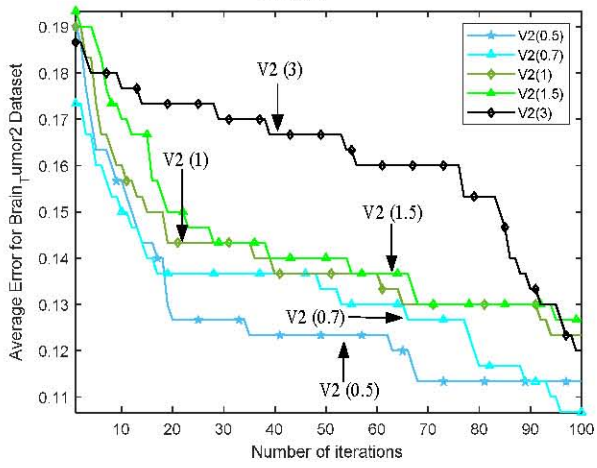
(l) Lymphoma



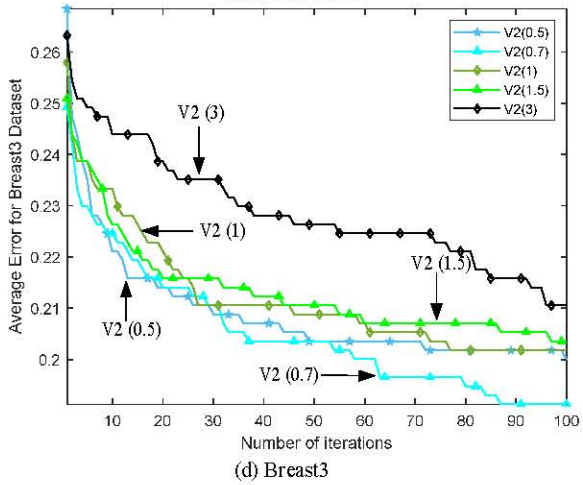
(a) Arrhythmia



(b) Brain Tumor1

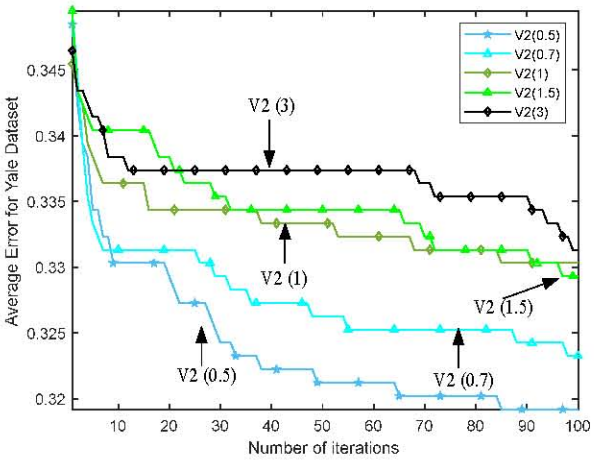


(c) Brain Tumor2

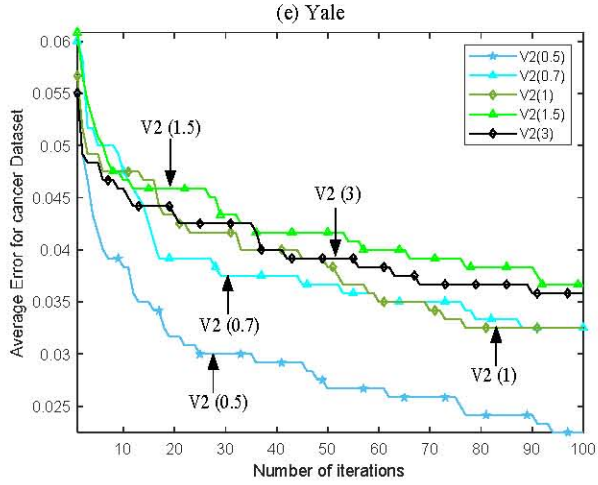


(d) Breast3

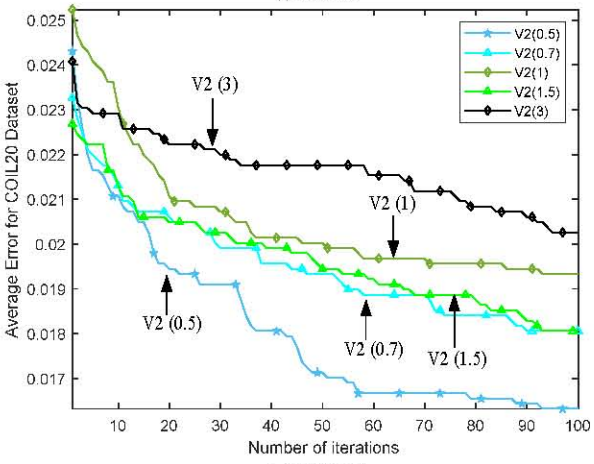
Fig. 4 Simulation results under V1-shaped transfer functions.



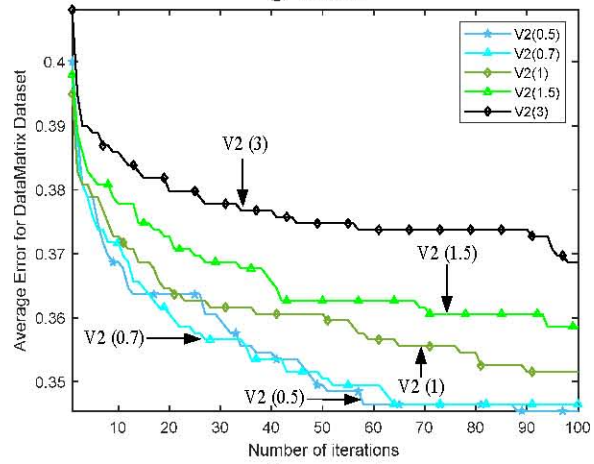
(e) Yale



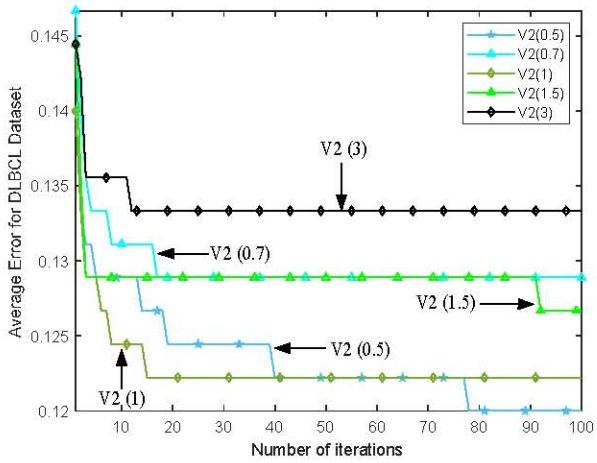
(f) Cancer



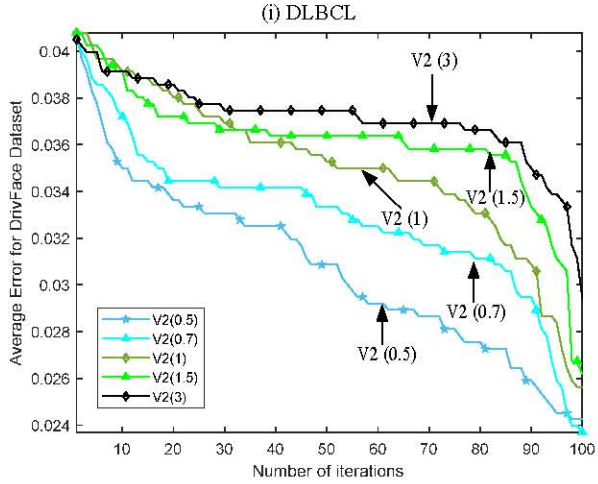
(g) COIL20



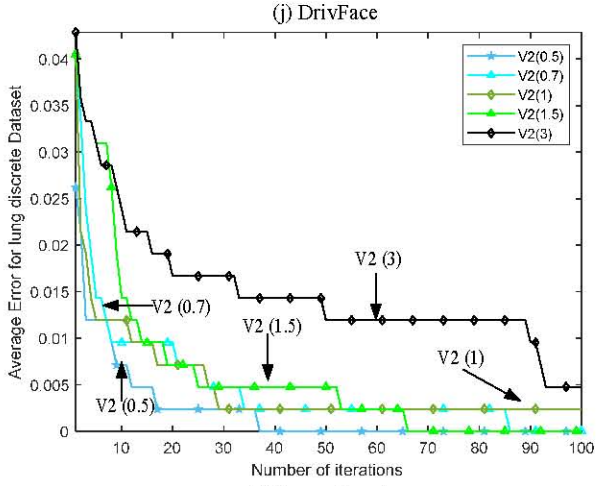
(h) DataMatrix



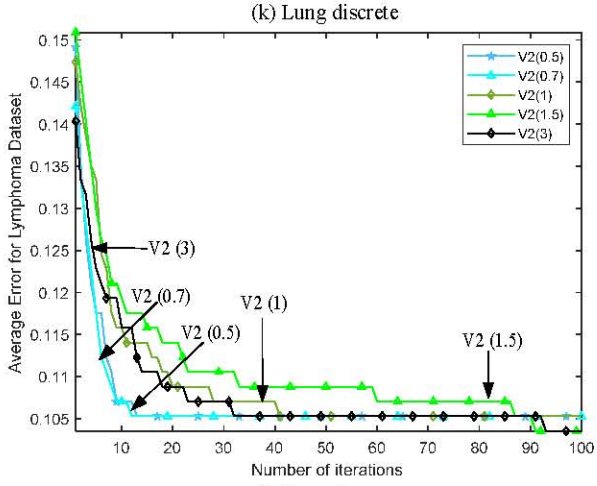
(i) DLBCL



(j) DrivFace

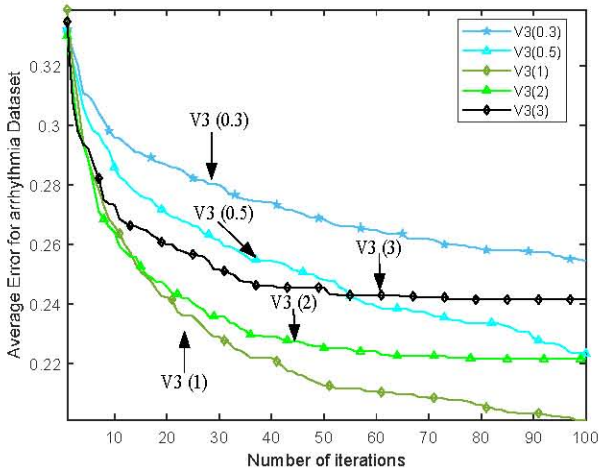


(k) Lung discrete

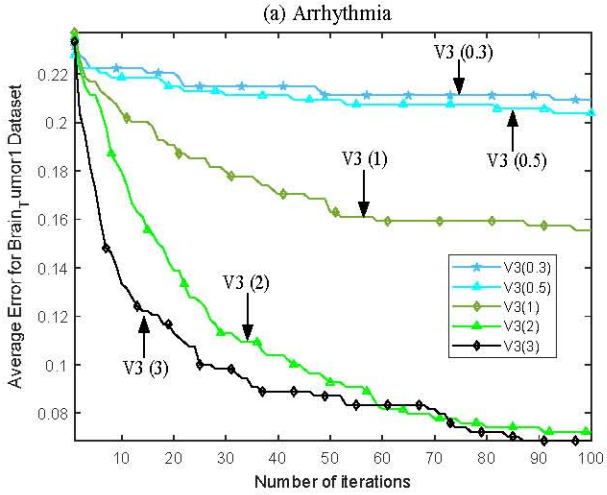


(l) Lymphoma

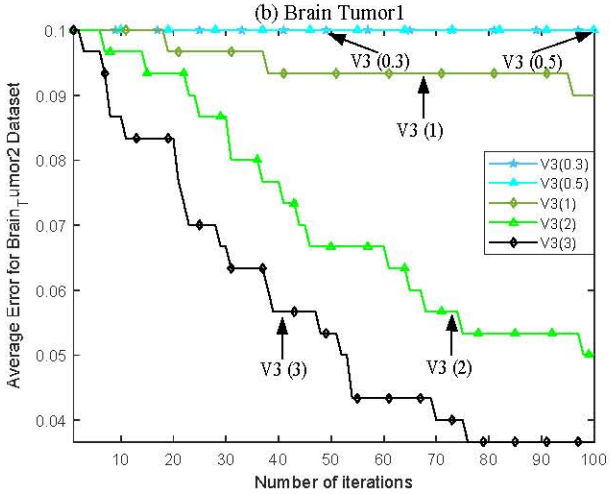
Fig. 5 Simulation results under V2-shaped transfer functions.



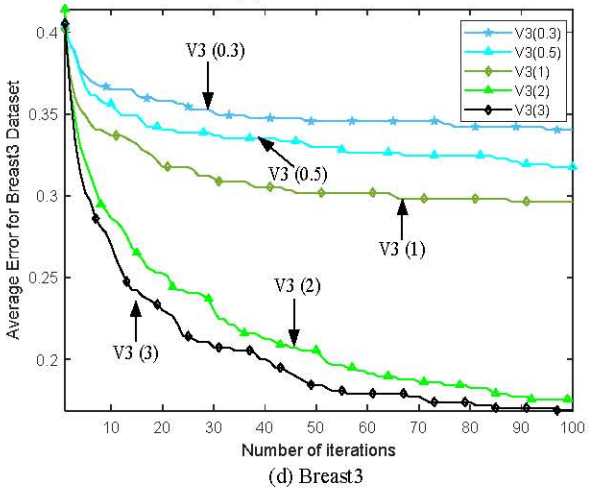
(a) Arrhythmia



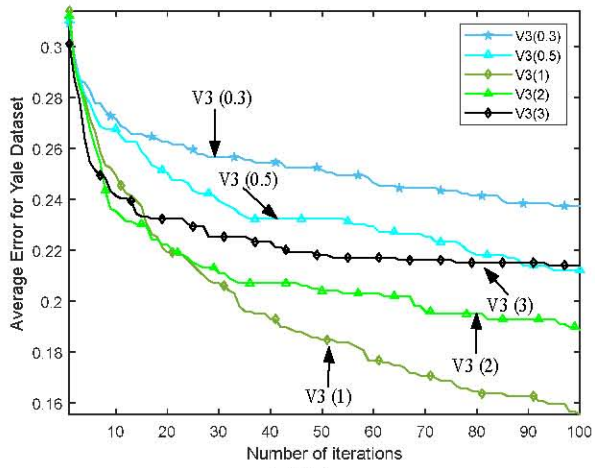
(b) Brain Tumor1



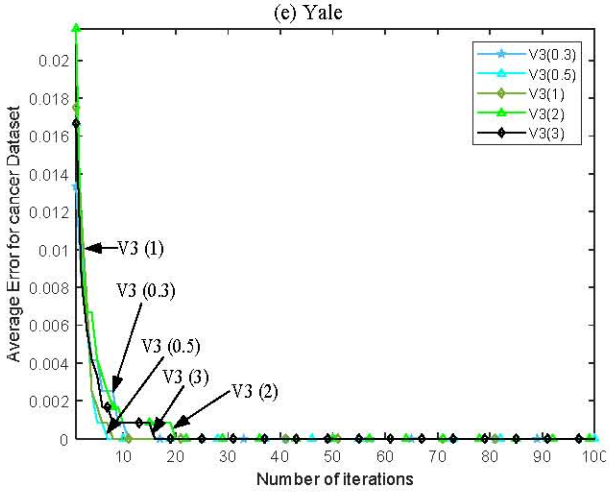
(c) Brain Tumor2



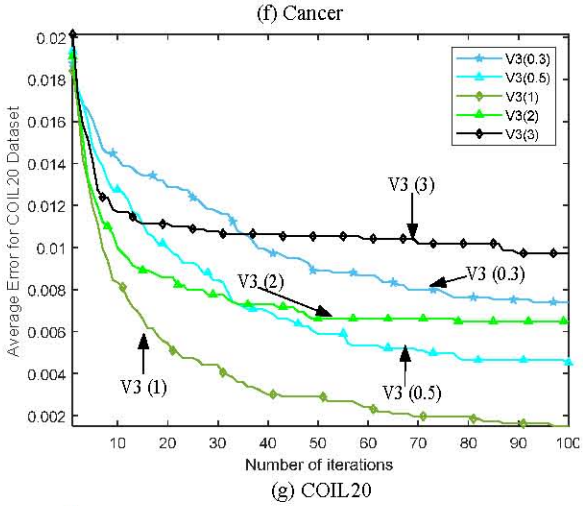
(d) Breast3



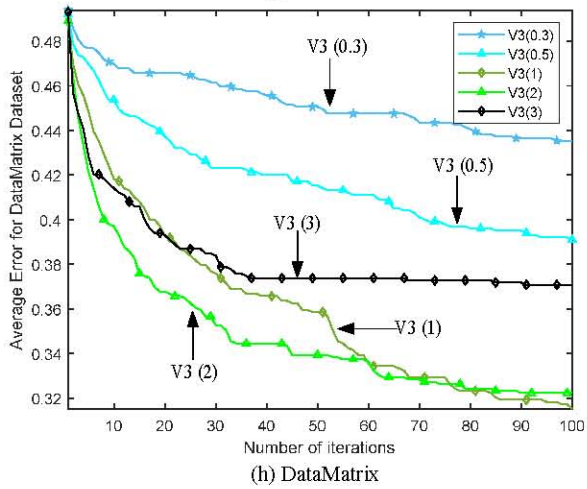
(e) Yale



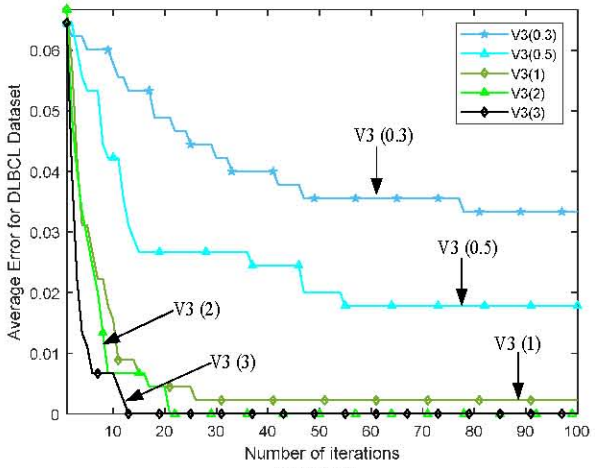
(f) Cancer



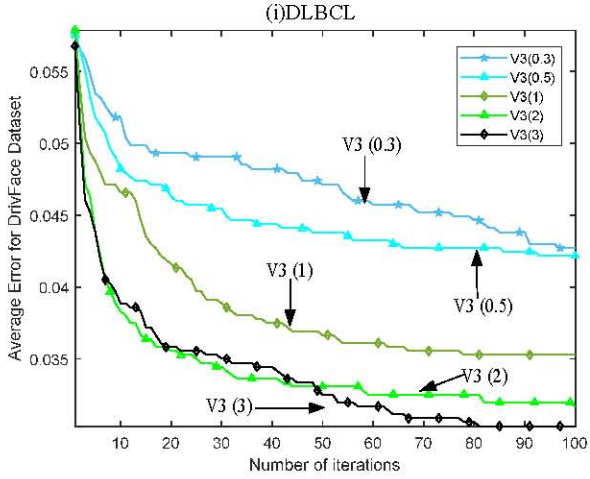
(g) COIL20



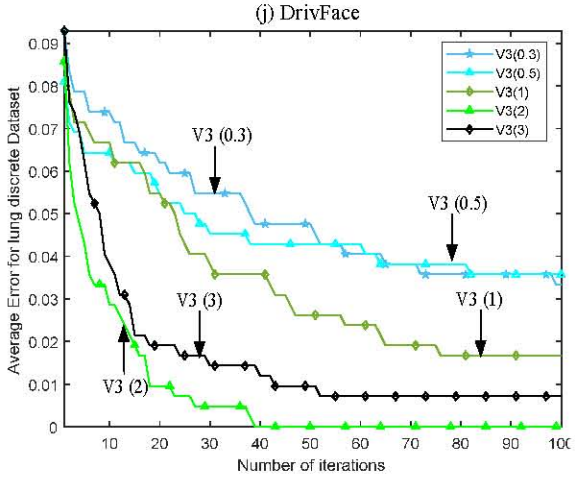
(h) DataMatrix



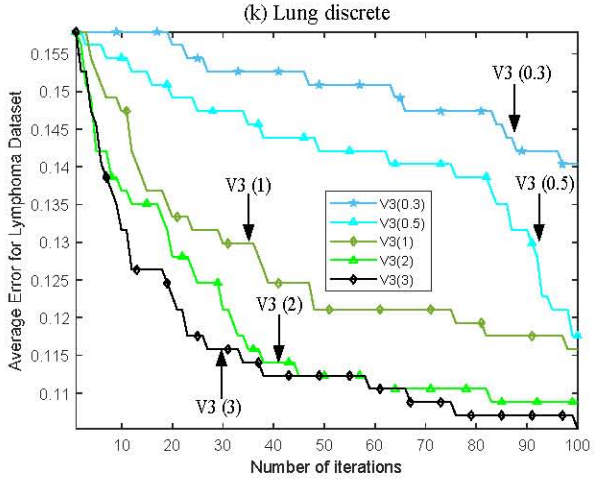
(i)DLBCL



(j) DrivFace

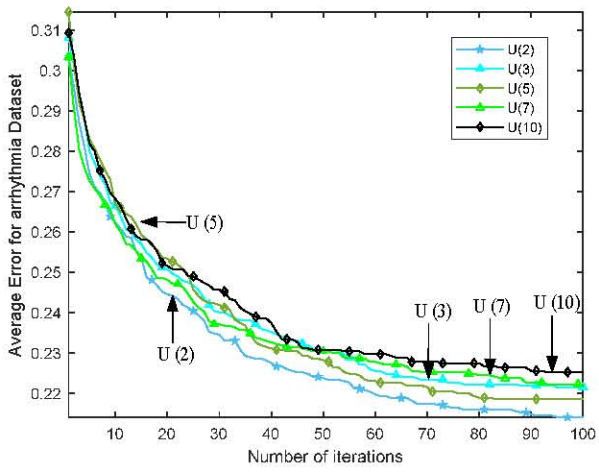


(k) Lung discrete

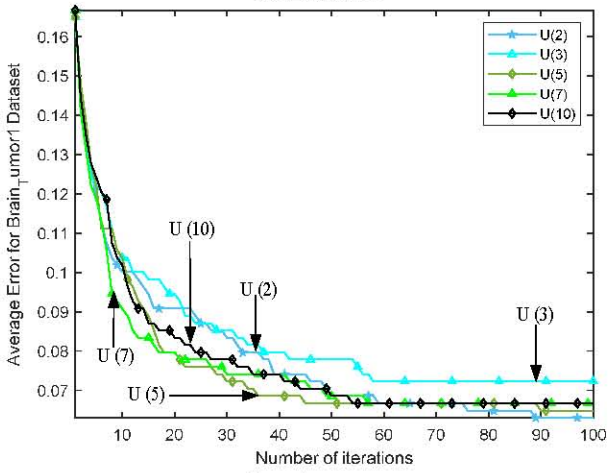


(l) Lymphoma

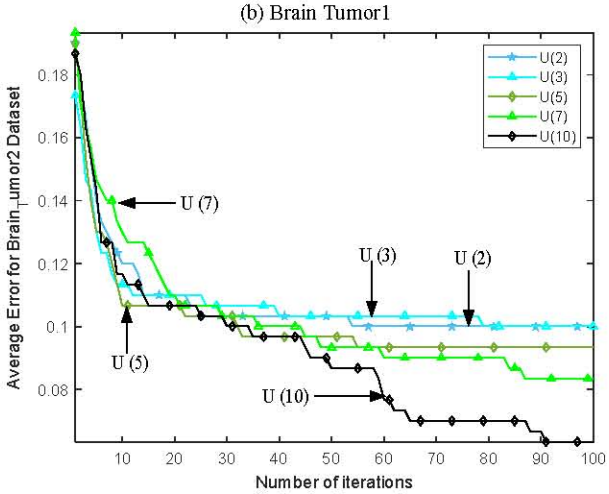
Fig. 6 Simulation results under V3-shaped transfer functions.



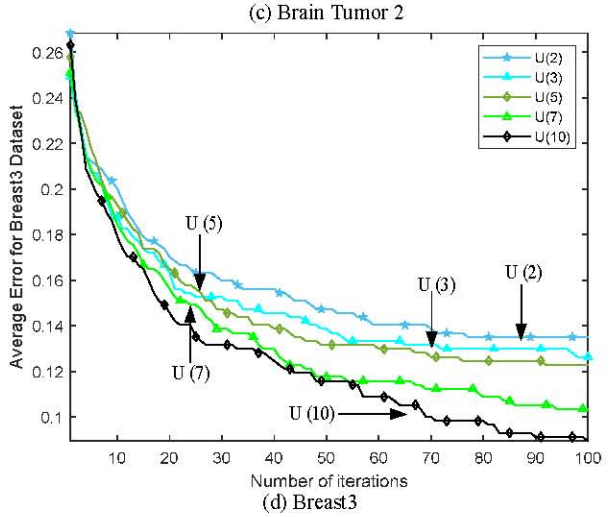
(a) Arrhythmia



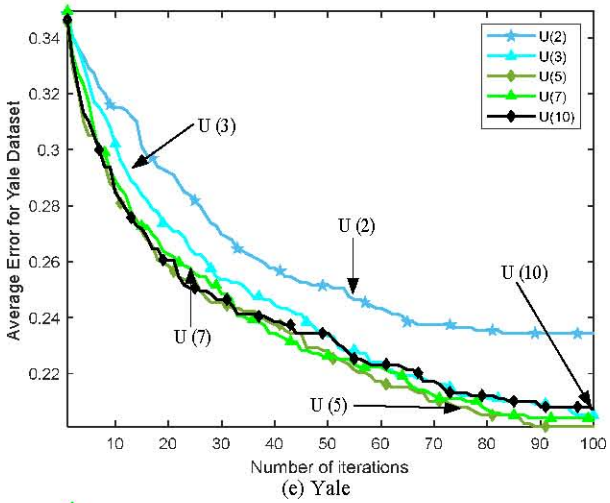
(b) Brain Tumor1



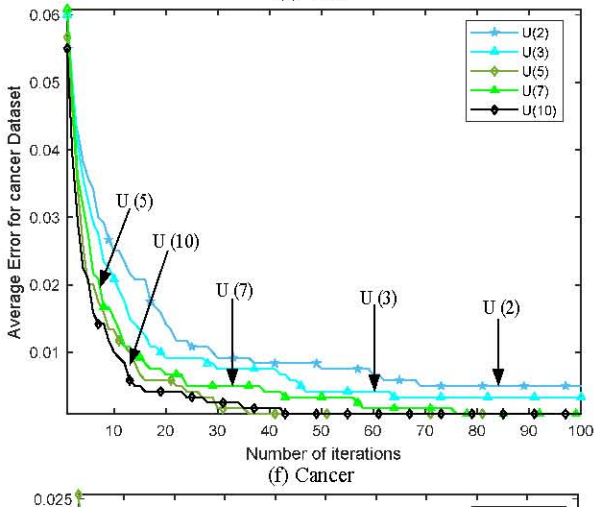
(c) Brain Tumor 2



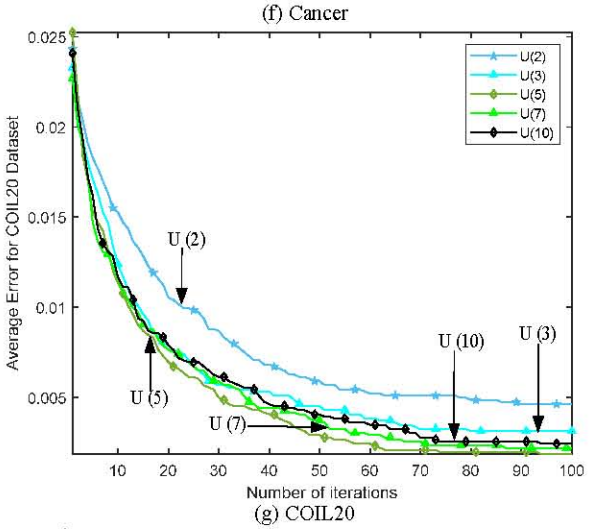
(d) Breast3



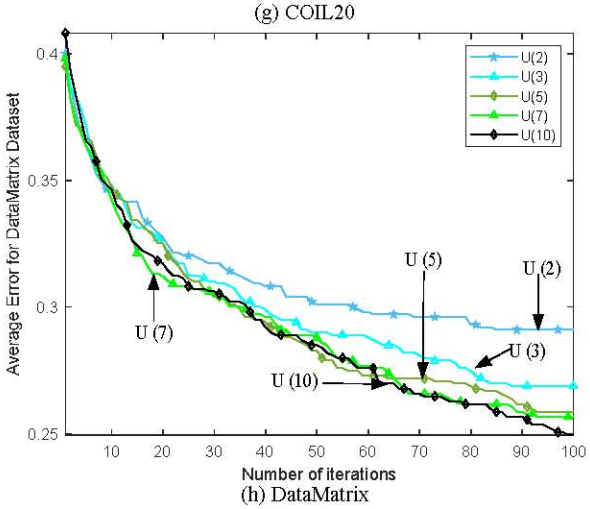
(e) Yale



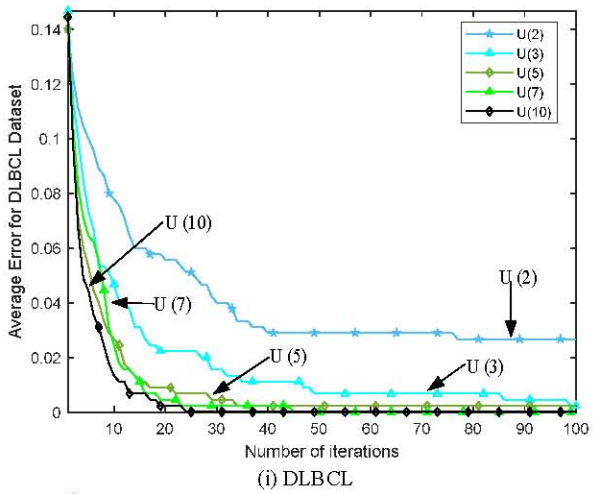
(f) Cancer



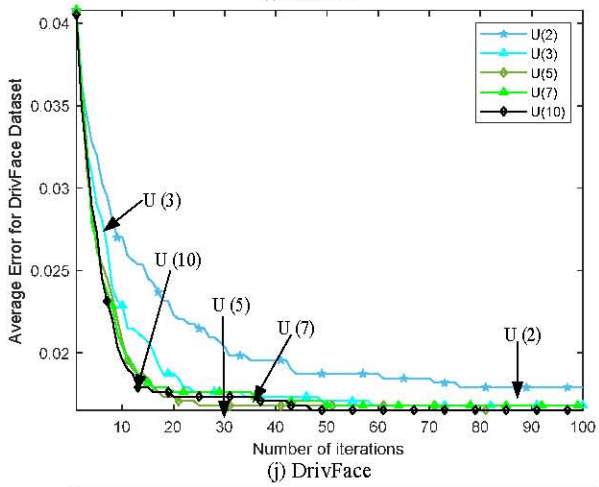
(g) COIL20



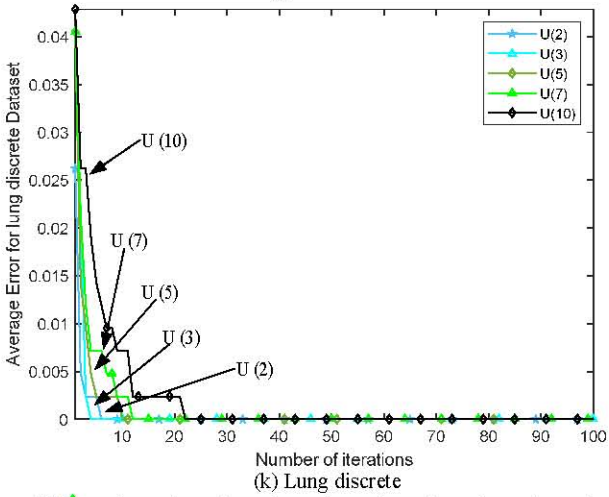
(h) DataMatrix



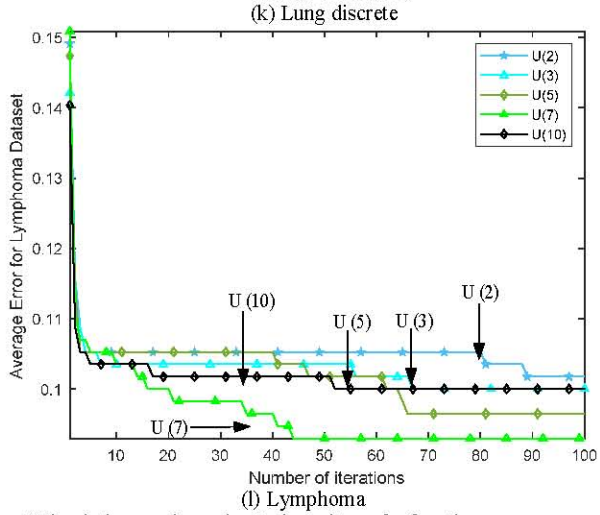
(i) DLBCL



(j) DrivFace

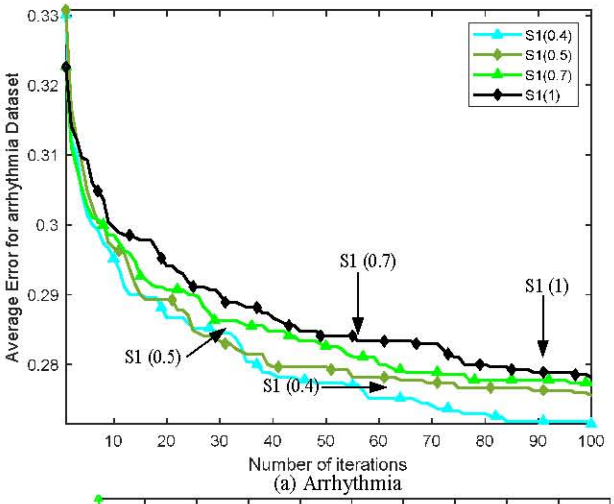


(k) Lung discrete

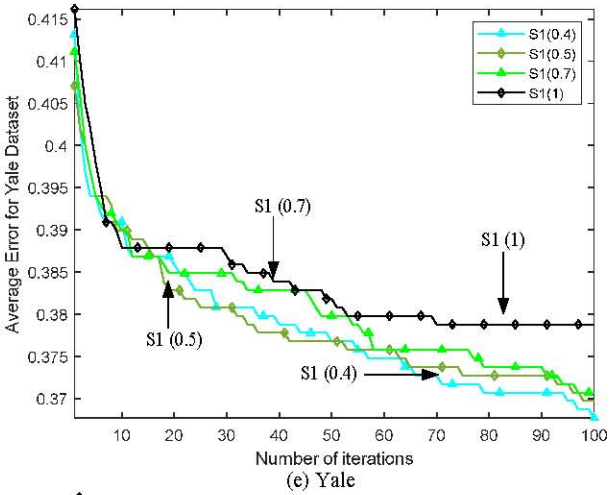


(l) Lymphoma

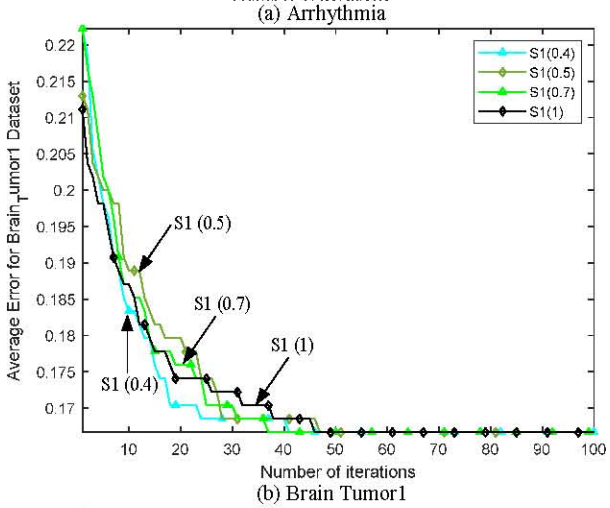
Fig. 7 Simulation results under U-shaped transfer functions.



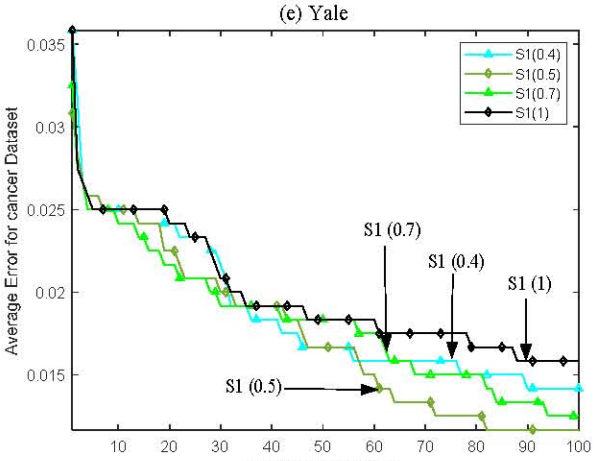
(a) Arrhythmia



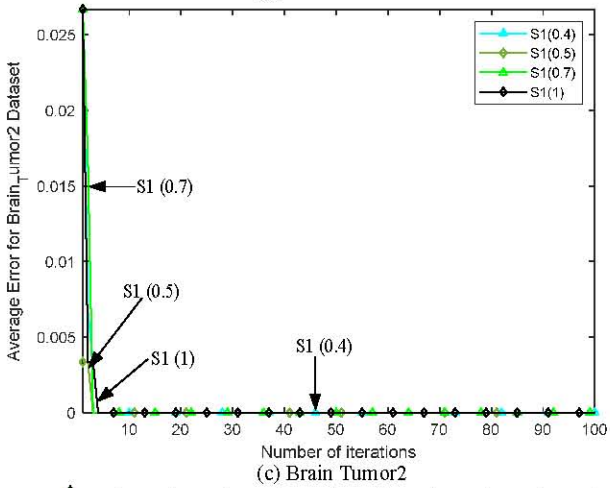
(e) Yale



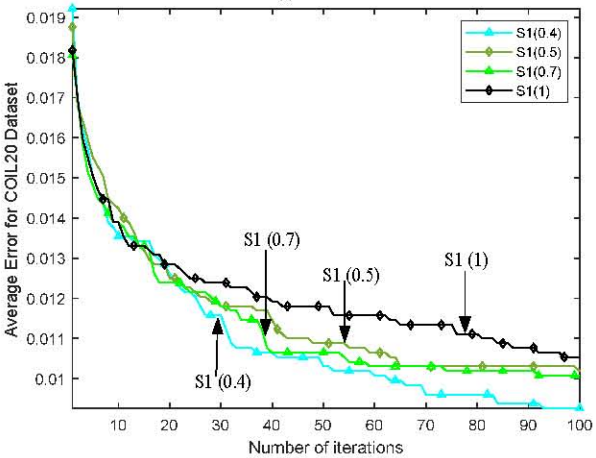
(b) Brain Tumor1



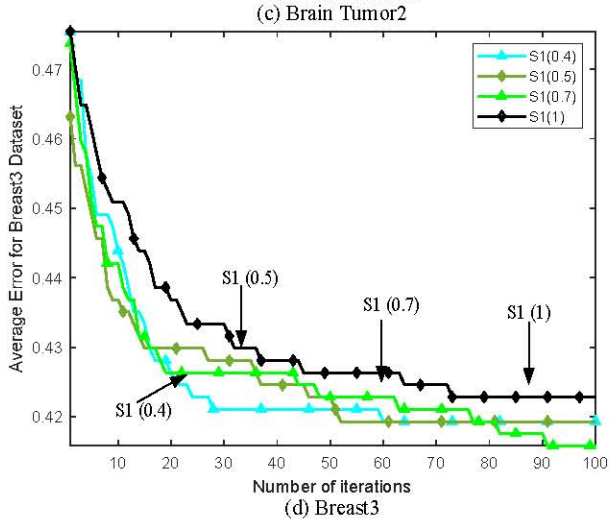
(f) Cancer



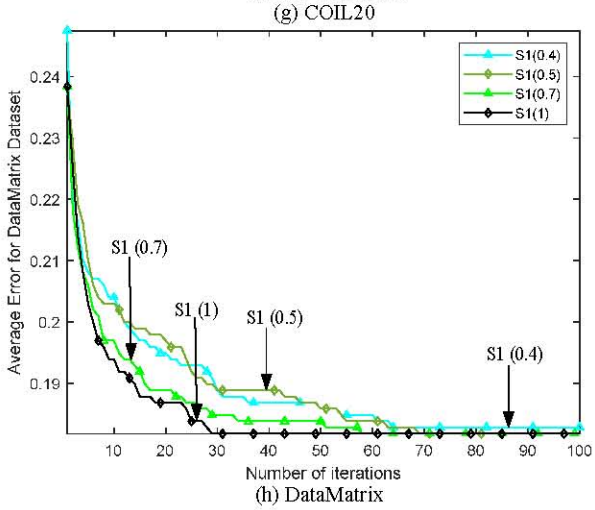
(c) Brain Tumor2



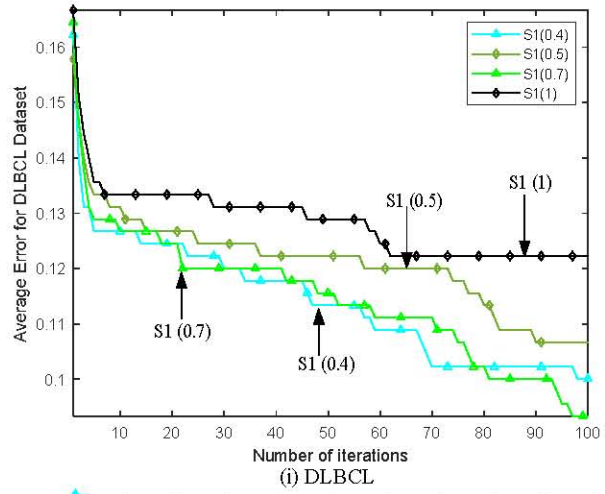
(g) COIL20



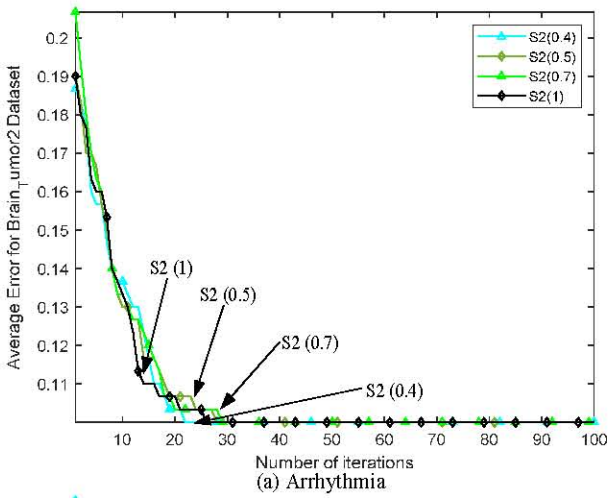
(d) Breast3



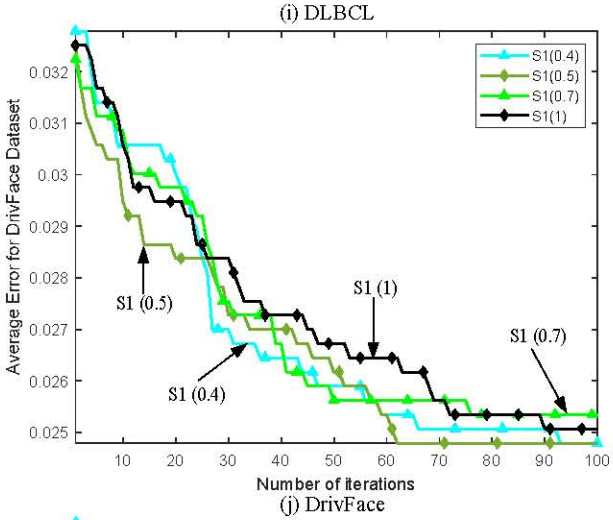
(h) DataMatrix



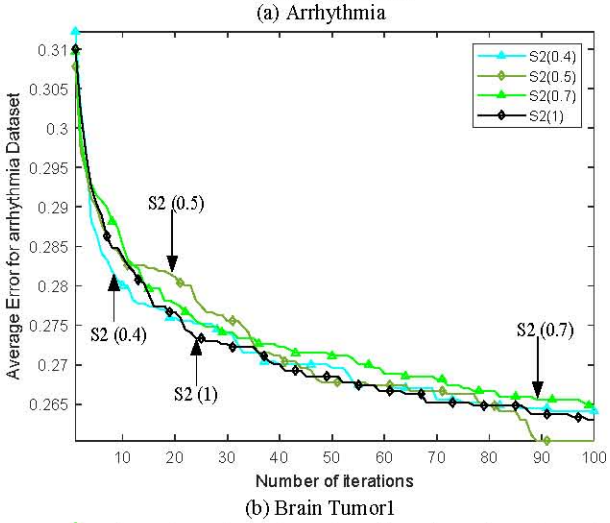
(i) DLBCL



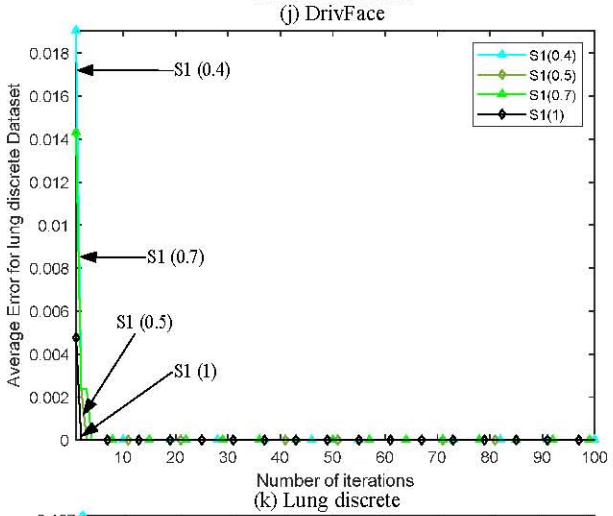
(a) Arrhythmia



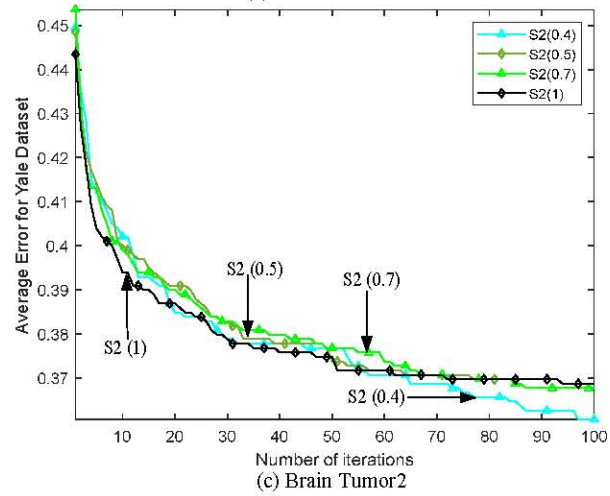
(j) DrivFace



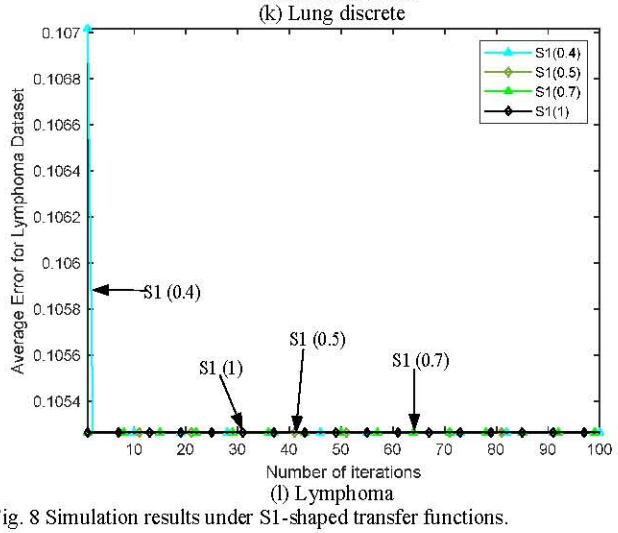
(b) Brain Tumor1



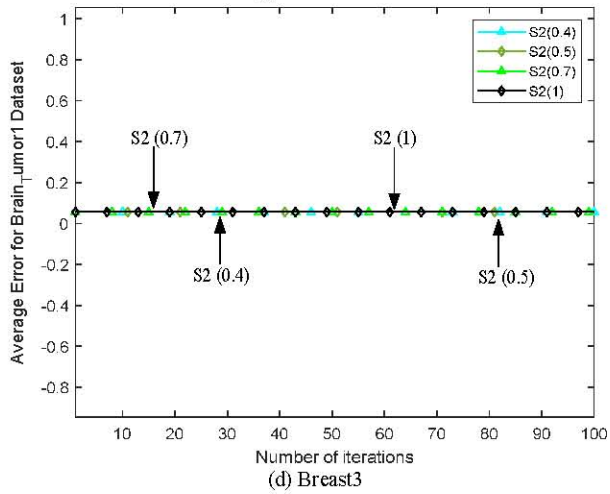
(k) Lung discrete



(c) Brain Tumor2

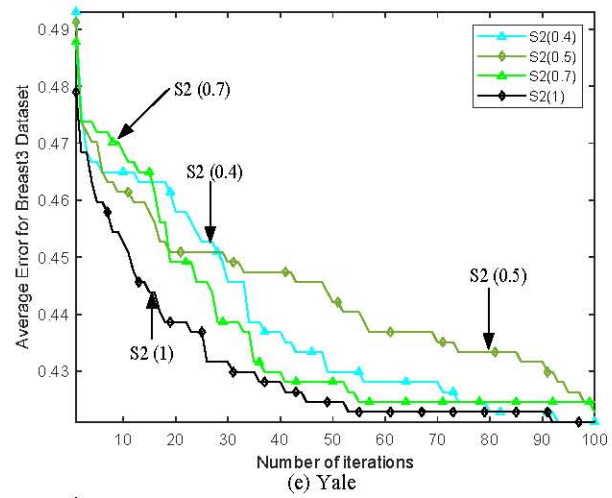


(l) Lymphoma

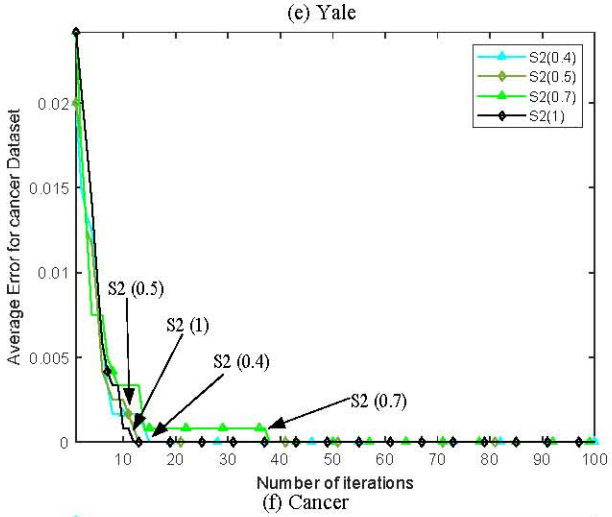


(d) Breast3

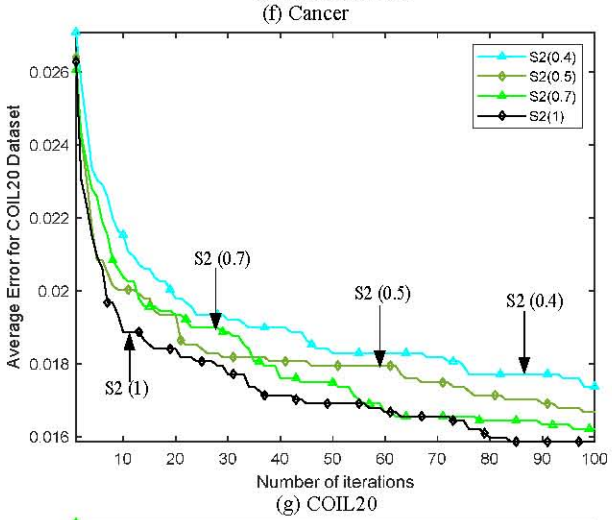
Fig. 8 Simulation results under S1-shaped transfer functions.



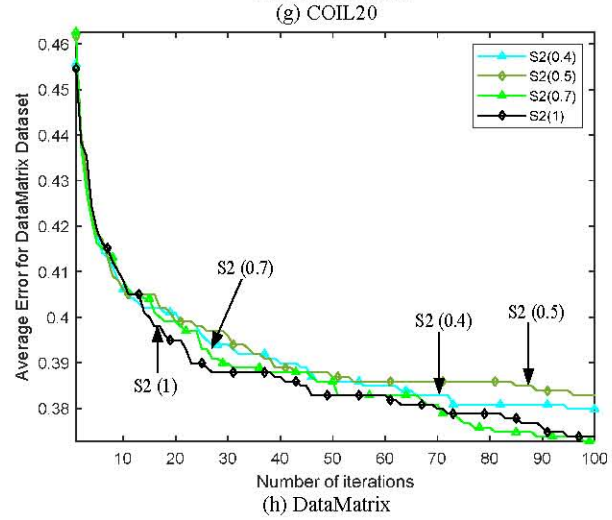
(e) Yale



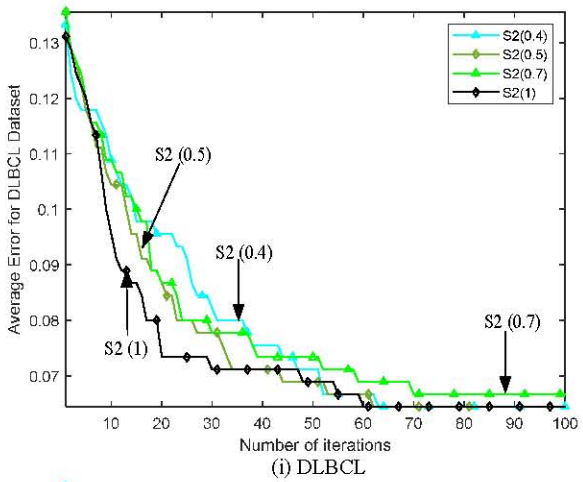
(f) Cancer



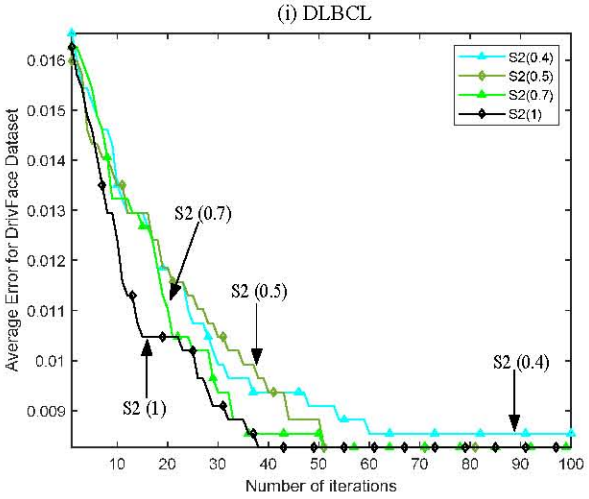
(g) COIL20



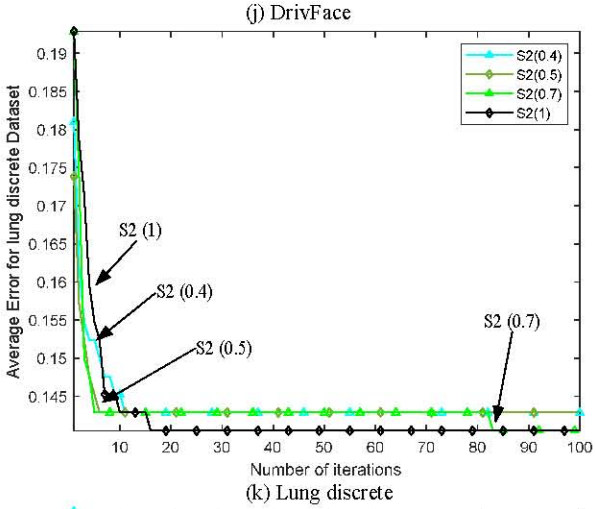
(h) DataMatrix



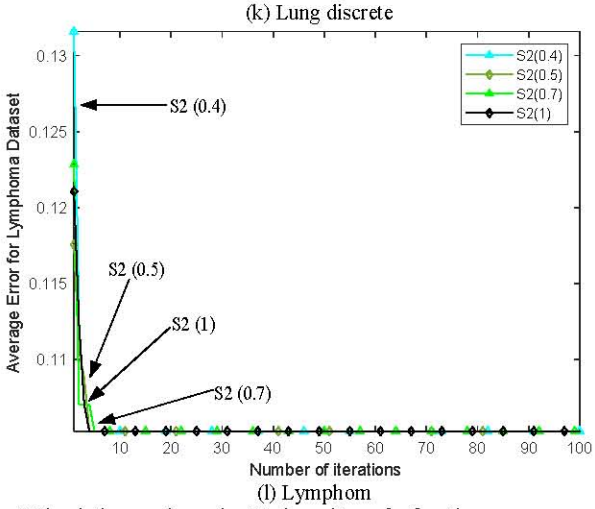
(i) DLBCL



(j) DrivFace

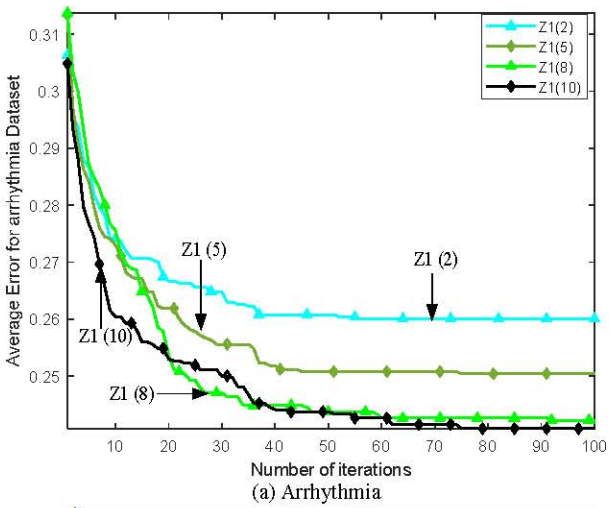


(k) Lung discrete

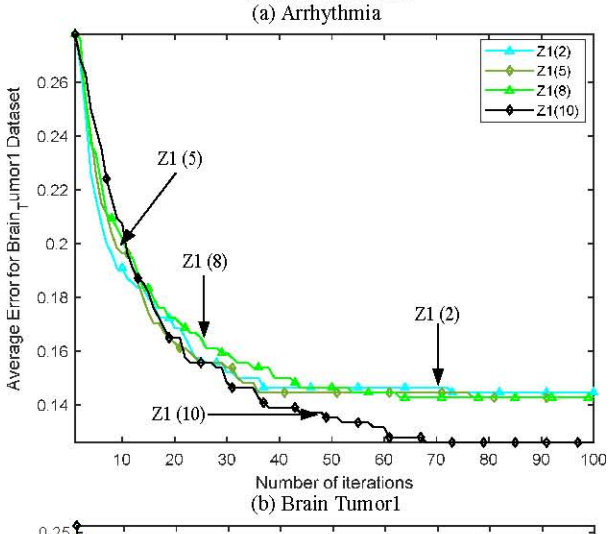


(l) Lymphom

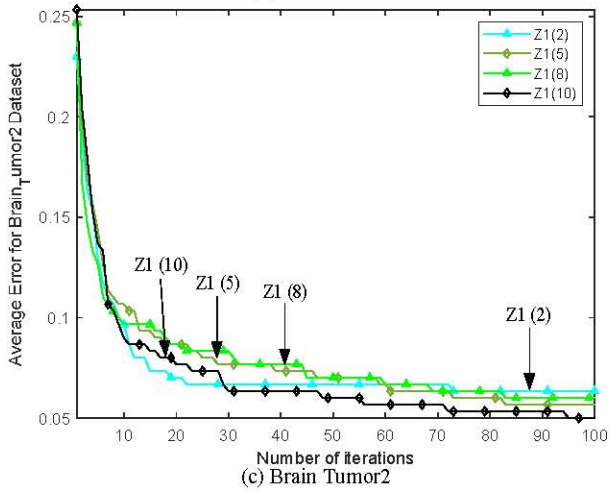
Fig. 9 Simulation results under S2-shaped transfer functions.



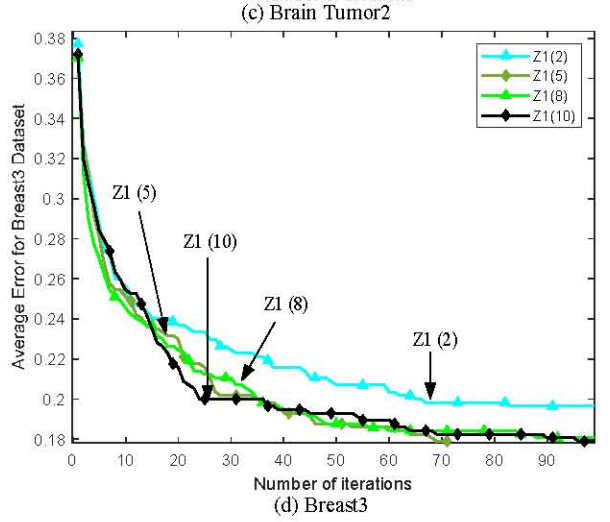
(a) Arrhythmia



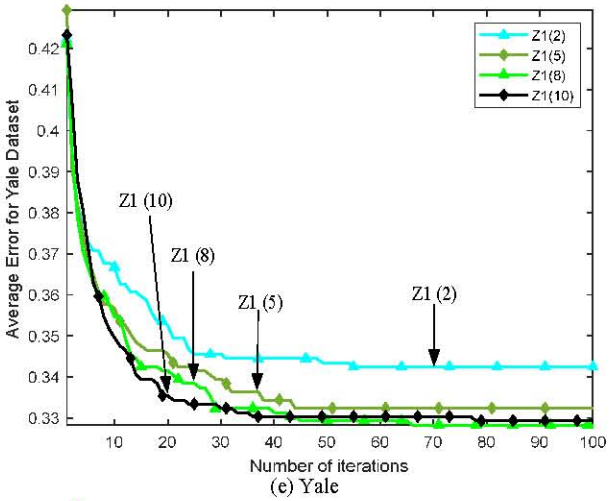
(b) Brain Tumor1



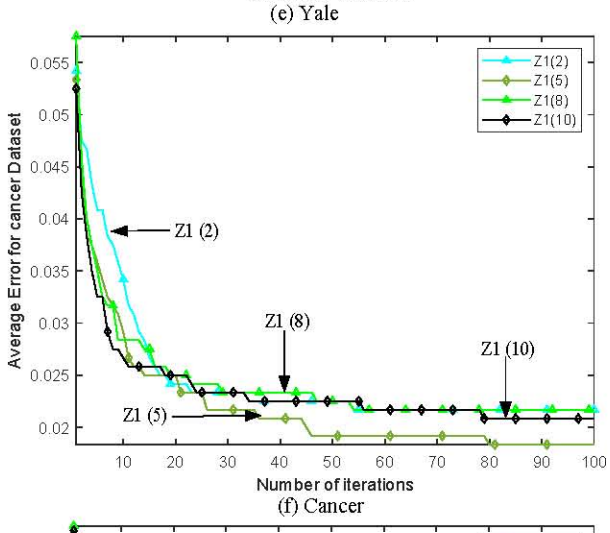
(c) Brain Tumor2



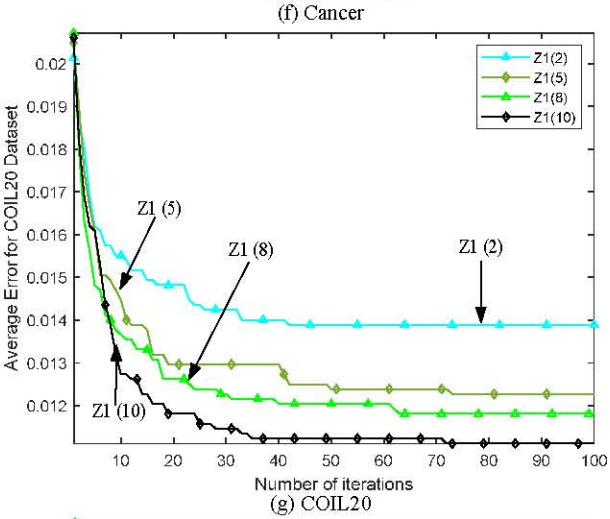
(d) Breast3



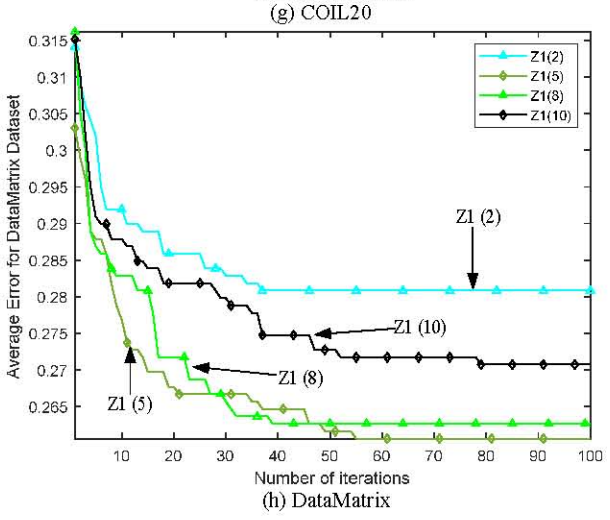
(e) Yale



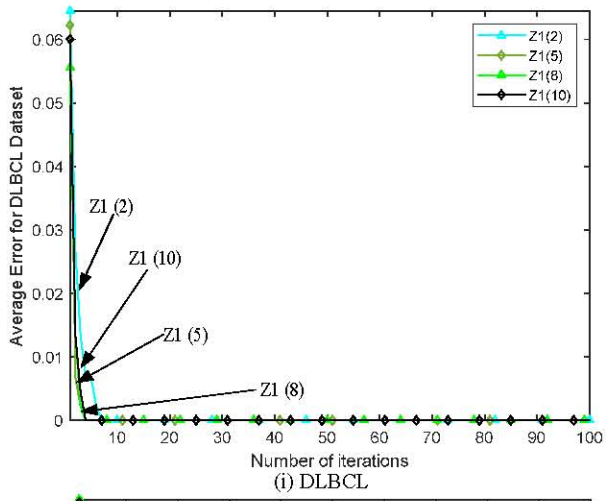
(f) Cancer



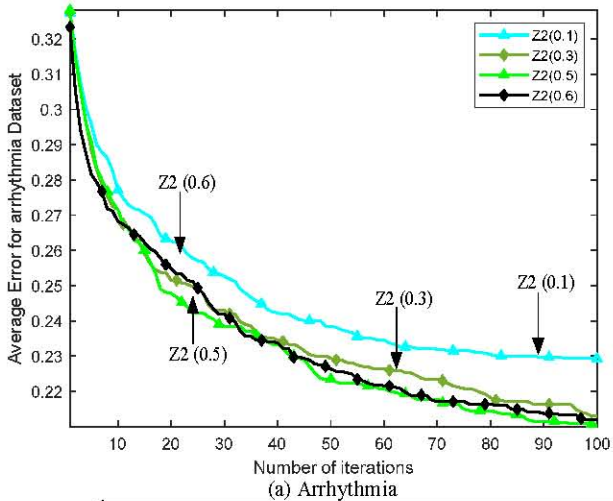
(g) COIL20



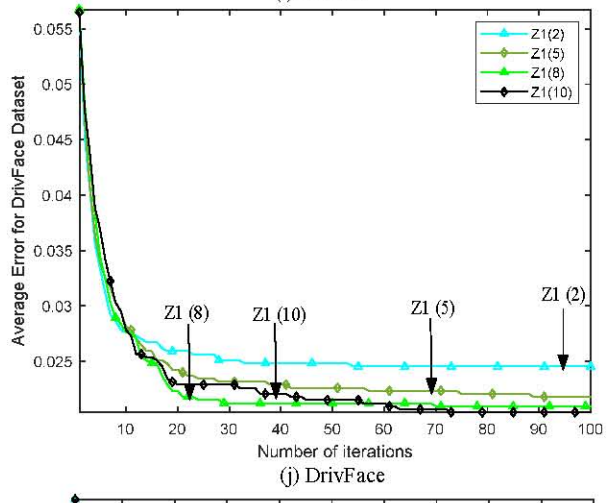
(h) DataMatrix



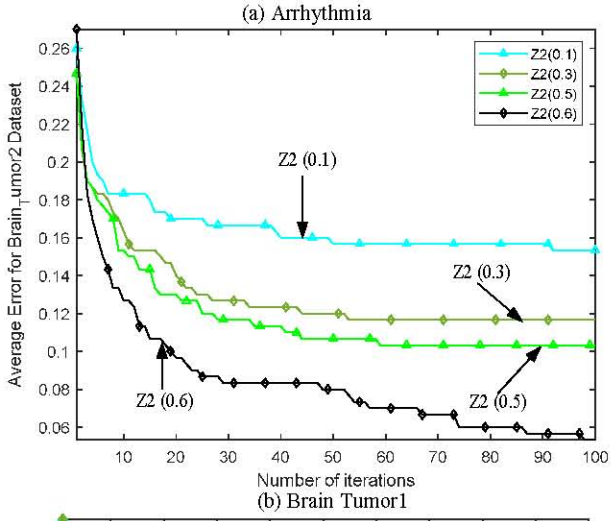
(i) DLBCL



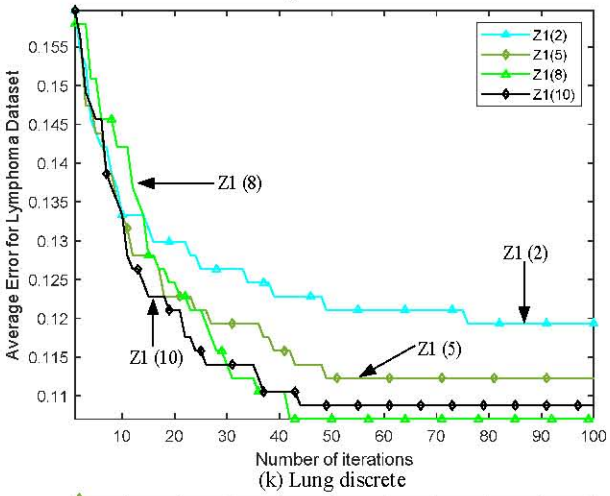
(a) Arrhythmia



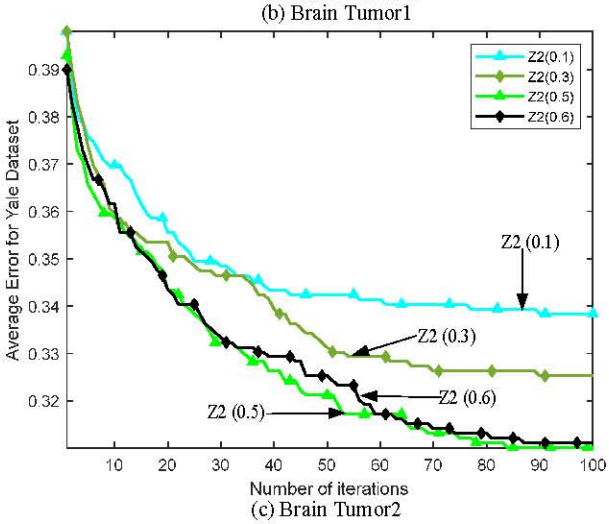
(j) DrivFace



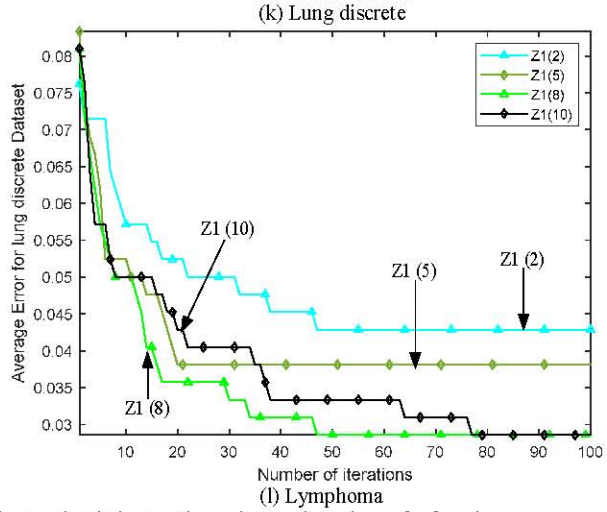
(b) Brain Tumor1



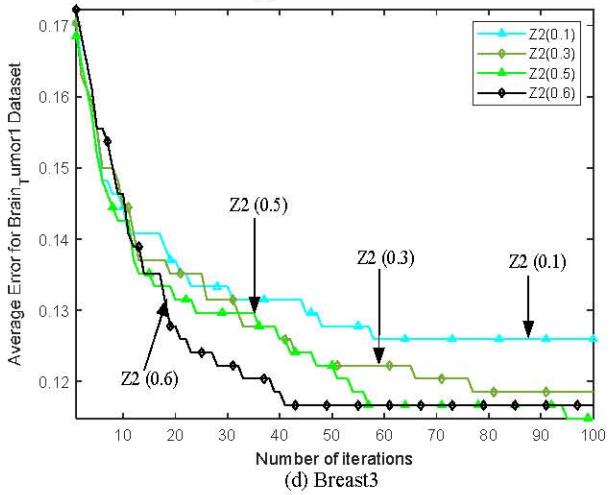
(k) Lung discrete



(c) Brain Tumor2

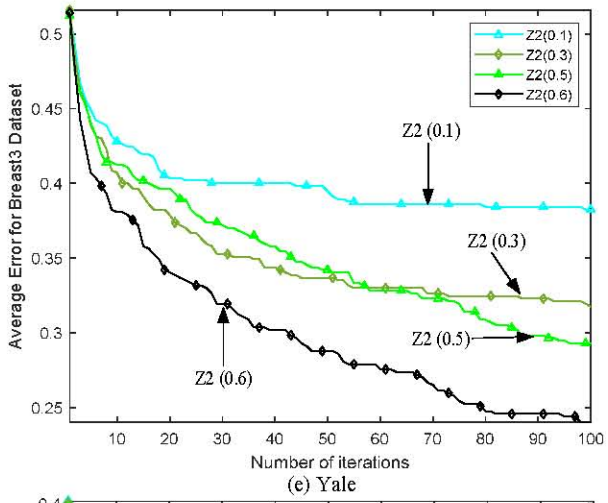


(l) Lymphoma

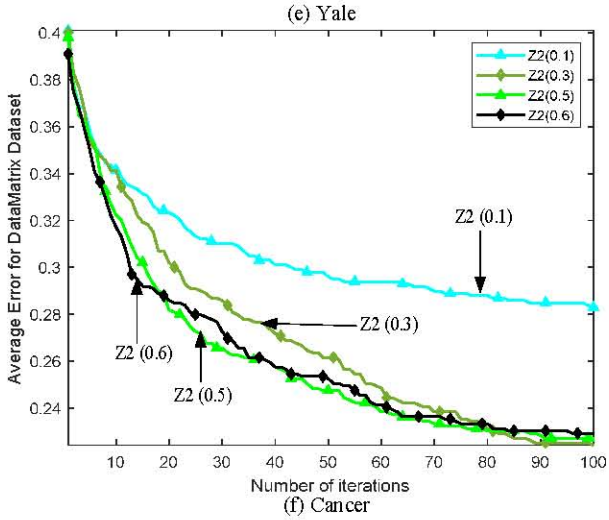


(d) Breast3

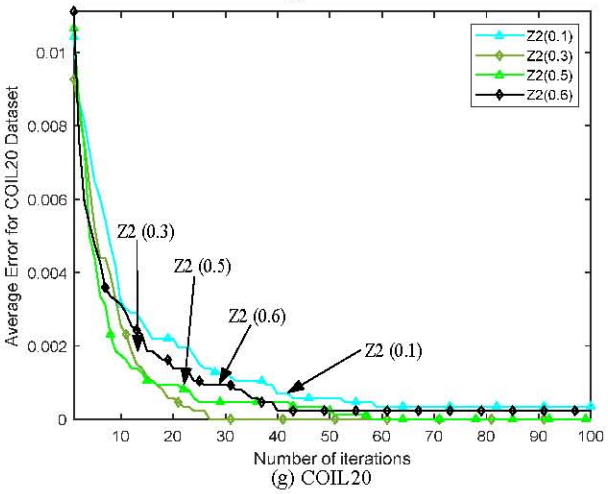
Fig. 10 Simulation results under Z1-shaped transfer functions.



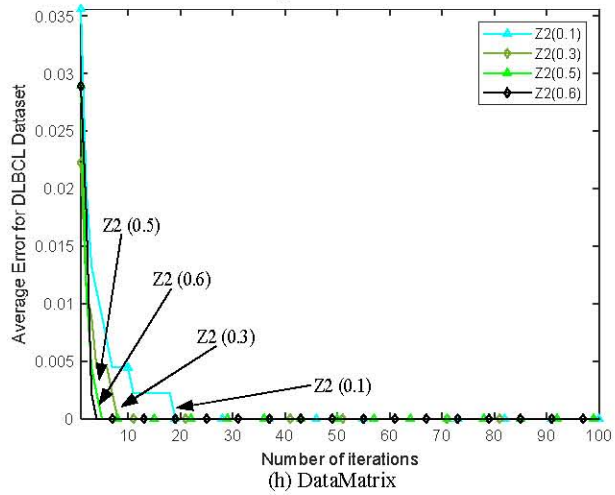
(e) Yale



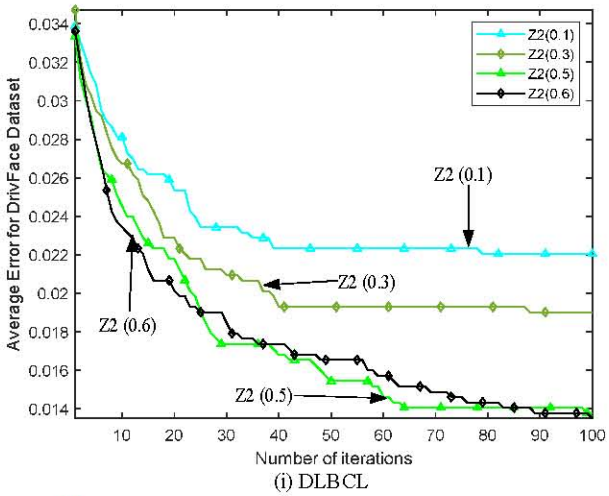
(f) Cancer



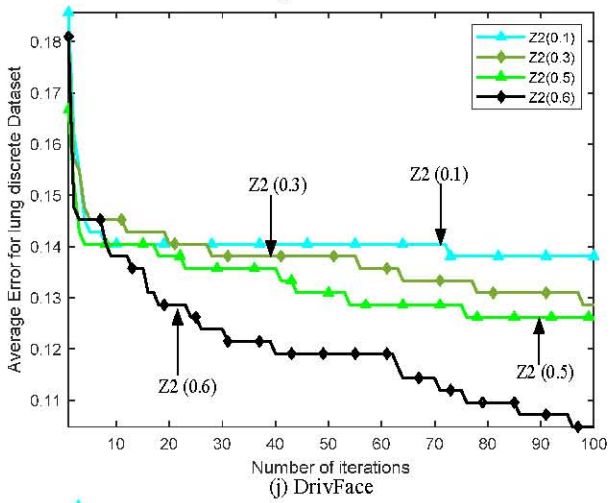
(g) COIL20



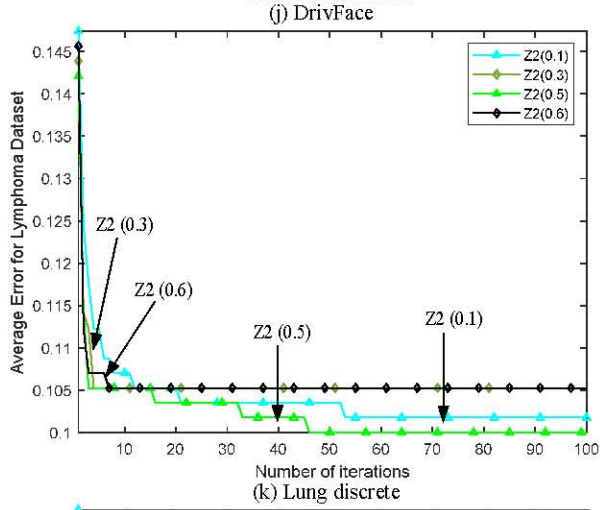
(h) DataMatrix



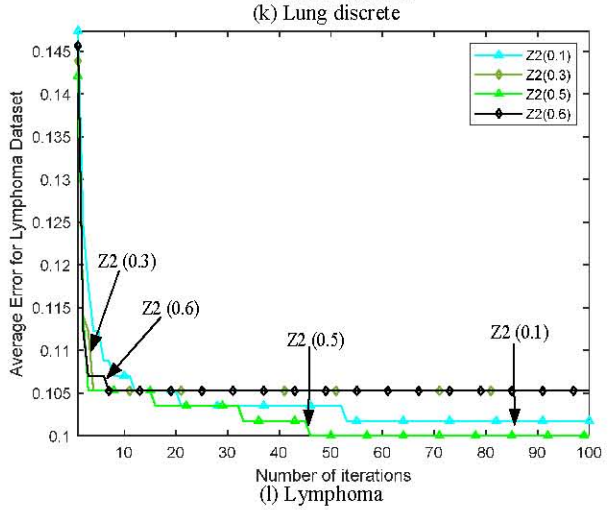
(i) DLBCL



(j) DrivFace

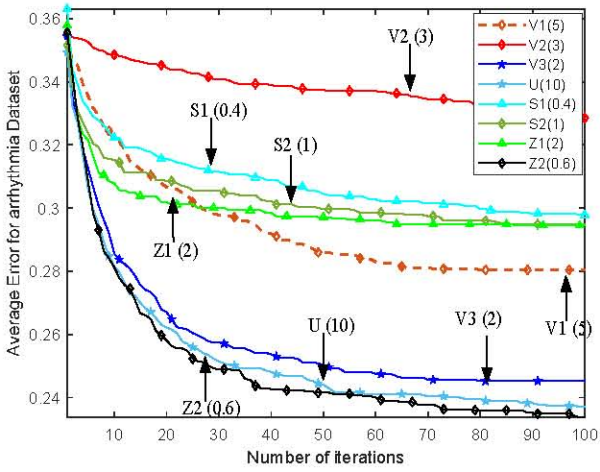


(k) Lung discrete

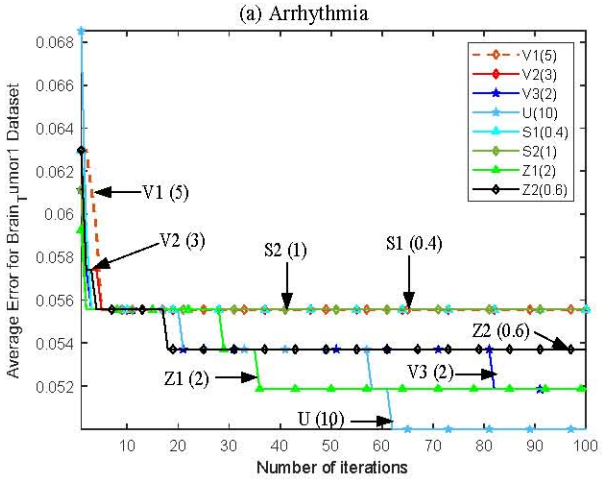


(l) Lymphoma

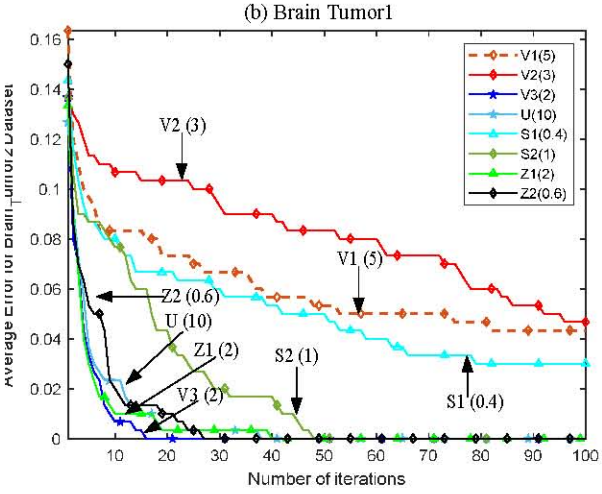
Fig. 11 Simulation results under Z2-shaped transfer functions.



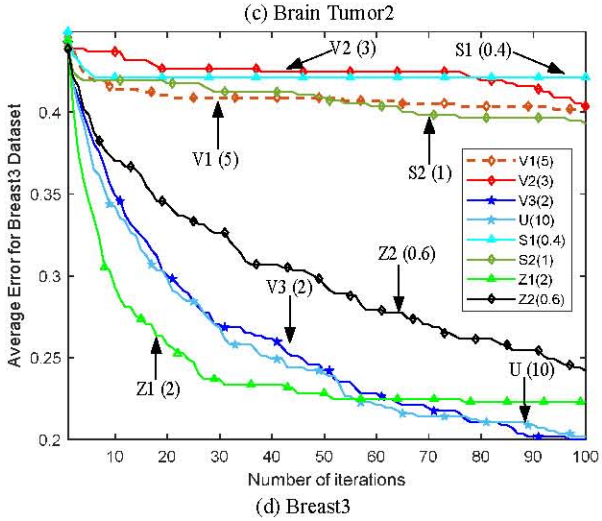
(a) Arrhythmia



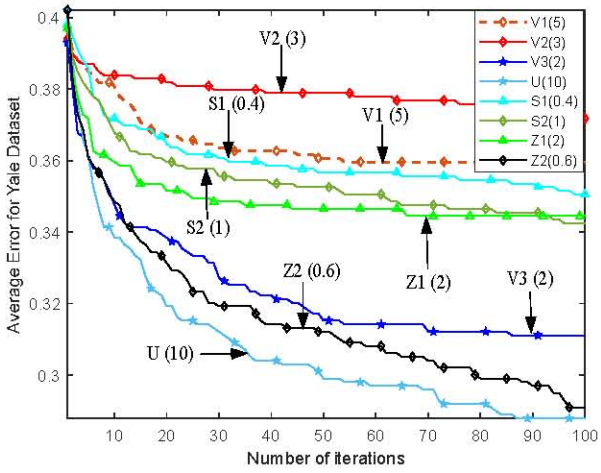
(b) Brain Tumor1



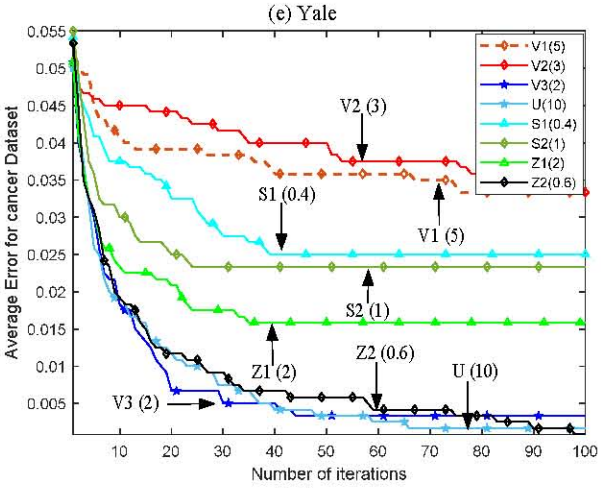
(c) Brain Tumor2



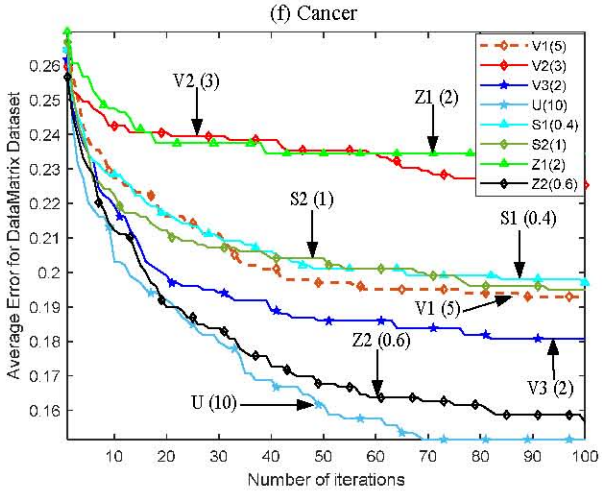
(d) Breast3



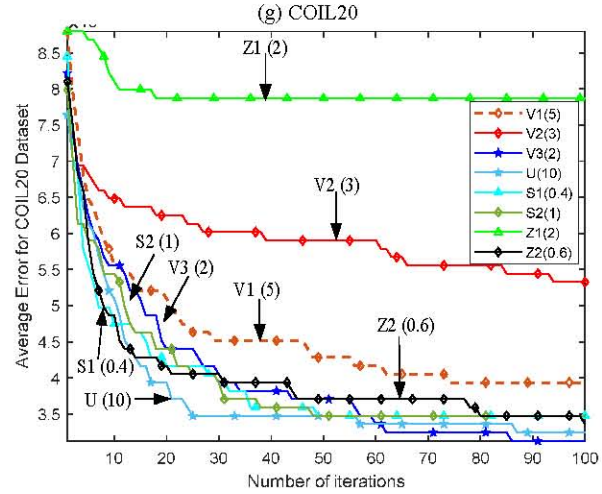
(e) Yale



(f) Cancer



(g) COIL20



(h) DataMatrix

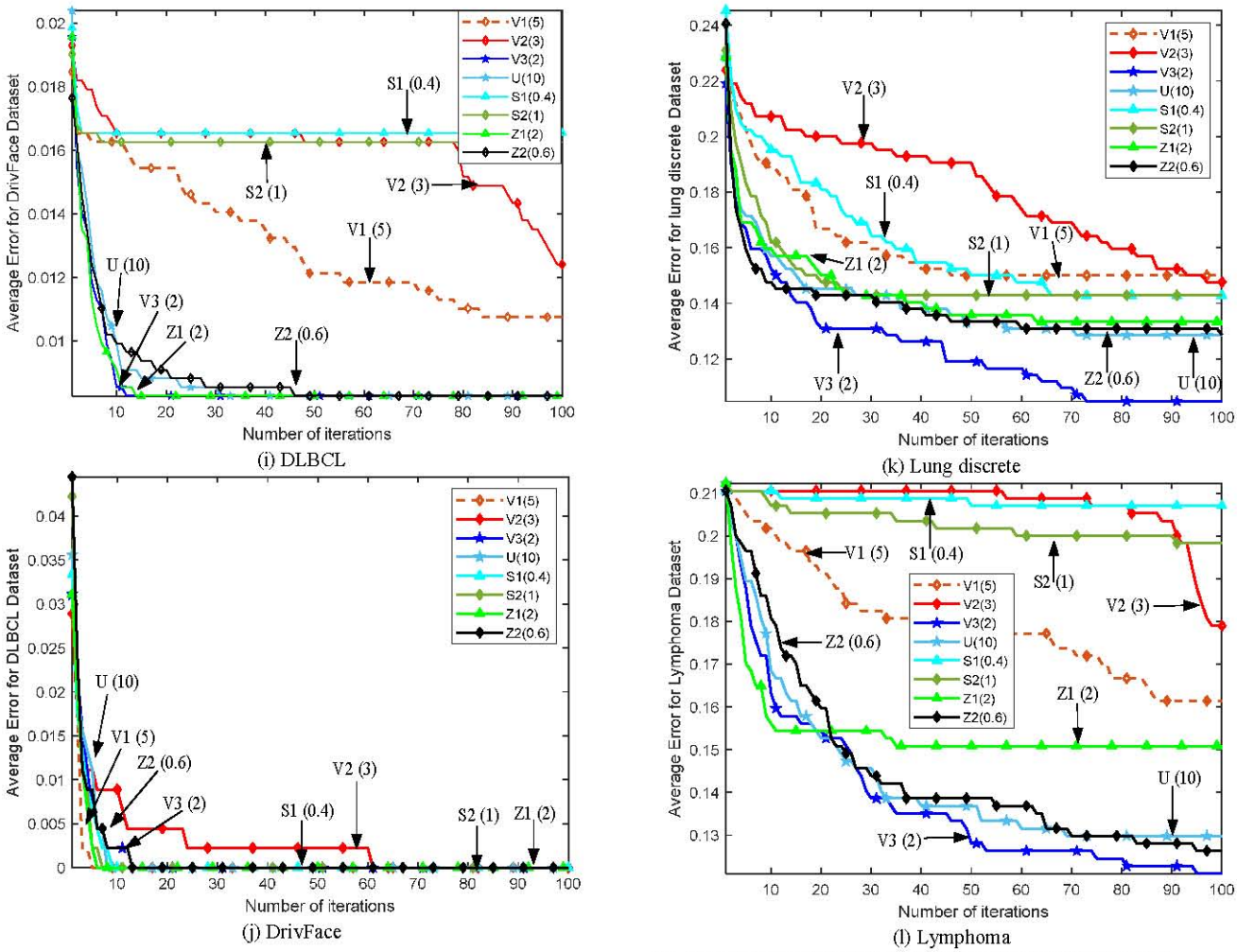


Fig. 12 Simulation results under eight kinds of transfer functions.

TABLE 3. AVERAGE AND STANDARD DEVIATION OF ACCURACY V1-SHAPED TRANSFER FUNCTIONS WITH DIFFERENT ORDER

Datasets	Measures	V1(0.1)	V1(0.5)	V1(1)	V1(3)	V1(5)
Arrhythmia	avg	0.7163	0.7059	0.6952	0.6948	**0.6867**
	std	0.0130	0.0133	0.0108	0.0126	**0.0089**
Brain Tumor1	avg	0.9000	0.9093	0.9019	0.8926	**0.8889**
	std	0.0226	0.0272	0.0239	0.0141	**0.0000**
Brain Tumor2	avg	0.9233	0.9200	0.9133	0.9100	**0.9067**
	std	0.0430	0.0407	0.0346	0.0305	**0.0254**
Breast3	avg	0.6842	0.6842	0.6825	0.6754	**0.6719**
	std	**0.0000**	**0.0000**	0.0096	0.0199	0.0226
Yale	avg	0.6909	0.6899	0.6879	0.6828	**0.6717**
	std	0.0147	0.0172	0.0213	0.0154	**0.0140**
Cancer	avg	**0.9992**	1.0000	1.0000	1.0000	**0.9992**
	std	0.0046	**0.0000**	**0.0000**	**0.0000**	0.0046
COIL20	avg	0.9983	0.9976	0.9968	0.9961	**0.9951**
	std	0.0018	**0.0016**	0.0022	0.0022	0.0022
DataMatrix	avg	0.6111	0.6081	0.5909	0.5859	**0.5788**
	std	0.0240	0.0238	0.0191	0.0216	**0.0122**
DLBCL	avg	0.9778	0.9733	0.9600	0.9756	**0.9356**
	std	0.0320	0.0332	0.0332	0.0327	**0.0122**
DrivFace	avg	0.9884	0.9876	0.9868	0.9851	**0.9835**
	std	0.0041	0.0042	0.0041	0.0034	**0.0000**
Lung discrete	avg	0.8976	0.8976	0.8929	0.8762	**0.8619**
	std	0.0360	0.0360	0.0409	0.0321	**0.0181**
Lymphoma	avg	0.8526	0.8509	0.8509	0.8456	**0.8421**
	std	0.0214	0.0199	0.0199	0.0134	**0.0000**

TABLE 4. AVERAGE CALCULATION TIME OF V1-SHAPED TRANSFER FUNCTIONS WITH DIFFERENT ORDER

Datasets	V1(0.1)	V1(0.5)	V1(1)	V1(3)	V1(5)
Arrhythmia	10.0064	9.7087	9.6858	9.8162	**9.6243**
Brain Tumor1	19.6963	18.5607	18.2900	17.9872	**16.7890**
Brain Tumor2	23.5781	22.1801	21.9086	22.0553	**21.6423**
Breast3	18.0015	17.4976	17.2110	16.8634	**16.5916**
Yale	9.2496	9.1835	8.9151	8.7876	**8.5487**
Cancer	75.4134	72.2311	68.2708	63.5299	**60.4801**
COIL20	144.5315	140.3265	135.7718	122.8624	**116.5039**
DataMatrix	8.9939	8.7488	8.6223	8.3254	**8.1656**
DLBCL	16.1390	15.3187	15.0190	14.9876	**14.5478**
DrivFace	128.6035	107.6543	99.6017	90.4255	**77.8444**
Lung discrete	5.2972	5.2899	5.2429	5.2513	**5.2413**
Lymphoma	14.3716	13.4724	12.9902	12.7698	**12.4384**

TABLE 5. AVERAGE NUMBER OF SELECTED FEATURES OF V1-SHAPED TRANSFER FUNCTIONS WITH DIFFERENT ORDER

Datasets	V1(0.1)	V1(0.5)	V1(1)	V1(3)	V1(5)
Arrhythmia	230.8333	221.6667	220.8000	211.9667	**207.0667**
Brain Tumor1	3302.8333	3159.7667	2819.0333	2696.8333	**2466.4667**
Brain Tumor2	5754.8000	5289.4000	5304.7000	**4960.0333**	5092.1000
Breast3	2861.5000	2798.9000	2727.7000	2649.9667	**2556.4667**
Yale	**629.5333**	709.7333	715.3333	641.0333	635.0667
Cancer	8853.6000	8206.7333	7645.3000	7106.5000	**6790.0333**
COIL20	653.2667	702.6333	701.3000	**642.6000**	648.2000
DataMatrix	**637.2333**	693.4667	699.1000	645.7333	643.3667
DLBCL	2709.1667	2511.4667	2733.7667	**2287.0667**	2516.2667
DrivFace	3204.4333	2761.5000	2392.1333	2065.2000	**1937.1000**
Lung discrete	166.1333	162.9000	150.6667	**141.4333**	149.2667
Lymphoma	2063.6000	1911.2667	1855.4333	1782.7000	**1746.4667**

TABLE 6. AVERAGE AND STANDARD DEVIATION OF ACCURACY V2-SHAPED TRANSFER FUNCTIONS WITH DIFFERENT ORDER

Datasets	Measures	V2(0.5)	V2(0.7)	V2(1)	V2(1.5)	V2(3)
Arrhythmia	avg	**0.6948**	0.7074	0.7074	0.7259	0.7307
	std	0.0191	0.0153	0.0153	**0.0125**	0.0188
Brain Tumor1	avg	0.7926	0.7981	0.7981	0.7944	**0.7870**
	std	0.0250	0.0399	0.0309	0.0390	**0.0211**
Brain Tumor2	avg	0.9033	0.9067	**0.9000**	0.9100	**0.9000**
	std	0.0183	0.0254	**0.0000**	0.0305	**0.0000**
Breast3	avg	0.6421	**0.6386**	0.6404	0.6456	0.6509
	std	0.0255	**0.0182**	0.0199	0.0237	0.0258
Yale	avg	**0.7293**	0.7313	0.7333	0.7434	0.7455
	std	0.0177	**0.0173**	0.0201	0.0206	0.0204
Cancer	avg	**1.0000**	**1.0000**	**1.0000**	**1.0000**	**1.0000**
	std	**0.0000**	**0.0000**	**0.0000**	**0.0000**	**0.0000**
COIL20	avg	**0.9843**	0.9868	0.9867	0.9880	0.9890
	std	0.0030	0.0032	0.0030	0.0028	**0.0027**
DataMatrix	avg	**0.5374**	0.5384	0.5495	0.5465	0.5586
	std	0.0177	0.0172	0.0248	**0.0169**	0.0272
DLBCL	avg	0.9622	0.9689	0.9667	**0.9600**	0.9756
	std	0.0336	0.0338	0.0339	0.0332	**0.0327**
DrivFace	avg	**0.9556**	0.9562	0.9565	0.9570	0.9562
	std	0.0046	**0.0044**	0.0048	0.0050	0.0049
Lung discrete	avg	0.9429	**0.9405**	0.9429	0.9452	0.9595
	std	0.0360	0.0307	0.0291	**0.0271**	0.0291
Lymphoma	avg	0.8614	0.8561	0.8526	0.8491	**0.8474**
	std	0.0258	0.0237	0.0214	0.0182	**0.0161**

TABLE 7. AVERAGE CALCULATION TIME OF V21-SHAPED TRANSFER FUNCTIONS WITH DIFFERENT ORDER

Datasets	V2(0.5)	V2(0.7)	V2(1)	V2(1.5)	V2(3)
Arrhythmia	6.9431	6.9080	**6.8045**	6.8504	6.8190
Brain Tumor1	16.3448	16.0910	15.2483	15.7220	**14.9512**
Brain Tumor2	18.6106	18.3511	**16.9116**	18.0540	16.9700
Breast3	14.1108	13.9804	13.2109	13.5212	**12.7242**
Yale	7.2314	7.2280	6.9967	6.9877	**6.7970**
Cancer	56.5779	55.8413	52.2627	50.9473	**47.4751**
COIL20	77.8699	75.7297	72.9639	70.1432	**65.4979**
DataMatrix	7.1679	7.0557	6.7972	6.9010	**6.4669**
DLBCL	13.4571	13.2223	12.4550	12.9249	**12.2709**
DrivFace	71.8315	62.5805	59.8968	52.1910	**45.6085**
Lung discrete	4.6058	4.5879	4.5189	4.5213	**4.4959**
Lymphoma	11.5857	11.2496	10.5130	10.7811	**10.0240**

TABLE 8. AVERAGE NUMBER OF SELECTED FEATURES OF V2-SHAPED TRANSFER FUNCTIONS WITH DIFFERENT ORDER

Datasets	V2(0.5)	V2(0.7)	V2(1)	V2(1.5)	V2(3)
Arrhythmia	239.2667	236.7333	232.8000	233.5333	**219.6333**
Brain Tumor1	2511.8667	2682.4667	**2507.7667**	2621.8000	2641.0000
Brain Tumor2	5001.3333	4866.1667	5191.9333	**4683.3000**	5171.4333
Breast3	**2232.9333**	2447.0333	2462.7333	2331.0000	2295.9333
Yale	773.8667	795.0667	761.3000	733.7000	**674.8000**
Cancer	7899.6000	8072.7667	7552.2667	6910.1333	**6834.9667**
COIL20	716.8000	701.2667	673.3000	696.3333	**657.8000**
DataMatrix	660.1000	686.1000	635.1667	639.9000	**612.1667**
DLBCL	2283.7333	2108.3000	2109.8333	2387.4333	**2028.6333**
DrivFace	1041.2667	997.8333	1112.6000	1169.0667	**900.1000**
Lung discrete	166.4333	156.1333	152.8333	152.5333	**139.5000**
Lymphoma	1877.4000	1771.7000	1699.6333	1603.0667	**1403.6000**

TABLE 9. AVERAGE AND STANDARD DEVIATION OF ACCURACY V3-SHAPED TRANSFER FUNCTIONS WITH DIFFERENT ORDER

Datasets	Measures	V3(0.3)	V3(0.5)	V3(1)	V3(2)	V3(3)
Arrhythmia	avg	**0.7456**	0.7767	0.7989	0.7785	0.7585
	std	**0.0150**	0.0207	0.0158	0.0204	0.0239
Brain Tumor1	avg	**0.7907**	0.7963	0.8444	0.9278	0.9315
	std	**0.0239**	0.0337	0.0423	0.0465	0.0348
Brain Tumor2	avg	**0.9000**	**0.9000**	0.9100	0.9500	0.9633
	std	**0.0000**	**0.0000**	0.0305	0.0509	0.0490
Breast3	avg	**0.6596**	0.6825	0.7035	0.8246	0.8316
	std	**0.0267**	0.0293	0.0293	0.0444	0.0321
Yale	avg	**0.7626**	0.7879	0.8444	0.8111	0.7859
	std	0.0277	0.0252	0.0273	0.0283	**0.0224**
Cancer	avg	**1.0000**	**1.0000**	**1.0000**	**1.0000**	**1.0000**
	std	**0.0000**	**0.0000**	**0.0000**	**0.0000**	**0.0000**
COIL20	avg	0.9926	0.9955	0.9985	0.9935	**0.9903**
	std	0.0031	0.0028	**0.0018**	0.0022	0.0033
DataMatrix	avg	**0.5646**	0.6091	0.6848	0.6778	0.6293
	std	**0.0292**	0.0359	0.0378	0.0313	0.0344
DLBCL	avg	**0.9667**	0.9822	0.9978	1.0000	1.0000
	std	0.0339	0.0300	0.0122	**0.0000**	**0.0000**
DrivFace	avg	**0.9573**	0.9579	0.9647	0.9683	0.9697
	std	0.0054	**0.0045**	0.0048	0.0049	**0.0045**
Lung discrete	avg	0.9667	**0.9643**	0.9833	1.0000	0.9929
	std	0.0362	0.0363	0.0307	**0.0000**	0.0218
Lymphoma	avg	**0.8596**	0.8825	0.8842	0.8912	0.8947
	std	0.0252	0.0226	0.0214	0.0134	**0.0000**

TABLE 10. AVERAGE CALCULATION TIME OF V3-SHAPED TRANSFER FUNCTIONS WITH DIFFERENT ORDER

Datasets	V3(0.3)	V3(0.5)	V3(1)	V3(2)	V3(3)
Arrhythmia	6.8220	5.9041	5.1100	**5.1073**	4.5718
Brain Tumor1	14.5373	13.1793	10.6973	9.2510	**9.1681**
Brain Tumor2	17.0242	15.8670	14.1953	**13.0417**	13.0771
Breast3	13.1032	11.6260	9.2628	**8.1621**	8.1925
Yale	6.6324	6.1226	5.4494	**5.1636**	5.2026
Cancer	48.5127	39.8851	28.9024	19.5624	**18.1206**
COIL20	59.9892	44.7737	19.3277	9.5177	**8.4365**
DataMatrix	6.6450	6.1271	5.4510	**5.1382**	5.2636
DLBCL	12.7328	11.3609	9.5328	**8.9365**	9.0310
DrivFace	48.5804	35.1392	20.5347	12.0890	**11.2807**
Lung discrete	4.6007	4.5226	4.4618	**4.3229**	4.4818
Lymphoma	10.4158	9.4589	8.2125	7.7338	**7.6499**

TABLE 11. AVERAGE NUMBER OF SELECTED FEATURES OF V3-SHAPED TRANSFER FUNCTIONS WITH DIFFERENT ORDER

Datasets	V3(0.3)	V3(0.5)	V3(1)	V3(2)	V3(3)
Arrhythmia	193.0333	114.6667	31.0667	10.0000	**6.8000**
Brain Tumor1	2818.0000	2581.0333	1523.8000	256.5333	**122.4333**
Brain Tumor2	5171.4333	5182.4667	4779.4667	2920.2000	**2179.7000**
Breast3	2178.1333	1914.6667	1185.6000	109.4667	**26.9000**
Yale	613.6667	500.5333	161.3667	71.1667	**68.9333**
Cancer	6698.9333	5954.9667	4929.9667	**3422.8000**	3644.6667
COIL20	613.0333	475.6333	189.5000	98.0000	**88.8000**
DataMatrix	610.4667	454.5667	147.5000	35.8333	**31.7667**
DLBCL	2373.7333	1989.7333	1410.6333	969.6333	**601.3667**
DrivFace	900.0667	983.1000	588.3667	169.8667	**89.2333**
Lung discrete	139.6667	130.5000	105.0667	51.6000	**37.3667**
Lymphoma	1605.3667	872.2333	848.3333	466.4667	**256.5667**

TABLE 12. AVERAGE AND STANDARD DEVIATION OF ACCURACY U-SHAPED TRANSFER FUNCTIONS WITH DIFFERENT ORDER

Datasets	Measures	U(2)	U(3)	U(5)	U(7)	U(10)
Arrhythmia	avg	0.7859	0.7785	0.7815	0.7778	**0.7748**
	std	0.0177	0.0184	0.0176	0.0177	**0.0160**
Brain Tumor1	avg	0.9370	**0.9278**	0.9352	0.9333	0.9333
	std	**0.0192**	0.0259	0.0211	0.0226	0.0226
Brain Tumor2	avg	**0.9000**	**0.9000**	0.9067	0.9167	0.9367
	std	**0.0000**	0.0263	0.0365	0.0379	0.0490
Breast3	avg	**0.8649**	0.8737	0.8772	0.8965	0.9105
	std	**0.0383**	0.0428	0.0422	0.0403	0.0501
Yale	avg	**0.7657**	0.7949	0.7990	0.7960	0.7929
	std	0.0355	0.0371	0.0377	0.0373	**0.0299**
Cancer	avg	**0.9950**	0.9967	0.9992	0.9992	0.9992
	std	0.0102	0.0086	**0.0046**	**0.0046**	**0.0046**
COIL20	avg	**0.9954**	0.9969	0.9981	0.9978	0.9976
	std	0.0039	0.0029	**0.0022**	0.0023	0.0023
DataMatrix	avg	**0.7091**	0.7313	0.7414	0.7434	0.7505
	std	**0.0334**	0.0371	0.0362	0.0335	0.0441
DLBCL	avg	**0.9733**	0.9978	0.9978	1.0000	1.0000
	std	0.0332	0.0122	0.0122	**0.0000**	**0.0000**
DrivFace	avg	**0.9821**	0.9832	0.9835	0.9832	0.9835
	std	0.0031	0.0015	**0.0000**	0.0015	**0.0000**
Lung discrete	avg	**1.0000**	**1.0000**	**1.0000**	**1.0000**	**1.0000**
	std	**0.0000**	**0.0000**	**0.0000**	**0.0000**	**0.0000**
Lymphoma	avg	**0.8982**	0.9000	0.9035	0.9070	0.9000
	std	**0.0134**	0.0161	0.0199	0.0226	0.0161

TABLE 13. AVERAGE CALCULATION TIME OF U-SHAPED TRANSFER FUNCTIONS WITH DIFFERENT ORDER

Datasets	U(2)	U(3)	U(5)	U(7)	U(10)
Arrhythmia	5.5289	5.1297	5.0550	5.0070	**4.8564**
Brain Tumor1	10.0219	10.0797	9.7364	9.5685	**9.4768**
Brain Tumor2	14.1522	14.2695	13.7683	13.6402	**13.4947**
Breast3	10.1158	9.8846	9.4230	9.1949	**8.9461**
Yale	5.7952	5.6056	5.4187	5.4363	**5.3170**
Cancer	29.1778	24.6007	21.8422	**20.1330**	20.1154
COIL20	23.9041	17.6930	12.8439	12.5754	**12.0118**
DataMatrix	5.8891	5.7141	5.5380	5.4698	**5.3890**
DLBCL	10.1041	9.7454	9.3311	9.3008	**9.1284**
DrivFace	20.3926	16.6078	14.1362	13.9091	**13.0630**
Lung discrete	4.4932	4.5083	**4.4825**	4.5327	4.4935
Lymphoma	2606.4689	8.8218	8.3243	**8.1449**	8.1906

TABLE 14. AVERAGE NUMBER OF SELECTED FEATURES OF U-SHAPED TRANSFER FUNCTIONS WITH DIFFERENT ORDER

Datasets	U(2)	U(3)	U(5)	U(7)	U(10)
Arrhythmia	97.2000	56.5333	47.0667	49.0000	**37.7333**
Brain Tumor1	836.0333	940.5000	662.3667	**445.3333**	470.2000
Brain Tumor2	2951.9667	3100.8333	2671.0667	**1441.0667**	1541.2667
Breast3	1449.1667	1170.9667	1033.0000	730.4000	**535.5333**
Yale	254.3000	165.6667	137.6333	120.3000	**106.1667**
Cancer	3928.6333	2578.1000	1995.4000	1861.2667	**1802.9000**
COIL20	214.6667	151.1000	100.4000	101.7667	**90.4333**
DataMatrix	363.5000	263.3667	182.1667	162.0000	**126.5333**
DLBCL	1425.0333	864.1000	618.4667	661.6667	**483.5667**
DrivFace	355.1000	255.8333	159.4667	180.1333	**141.4000**
Lung discrete	128.7000	115.9333	111.4000	112.3333	**102.3333**
Lymphoma	933.5333	1086.6333	707.8000	**690.2000**	947.0000

TABLE 15. AVERAGE AND STANDARD DEVIATION OF ACCURACY S1-SHAPED TRANSFER FUNCTIONS WITH DIFFERENT ORDER

Datasets	Measures	S1(0.4)	S1(0.5)	S1(0.7)	S1(1)
Arrhythmia	avg	0.7285	0.7244	0.7226	**0.7219**
	std	0.0081	**0.0074**	0.0099	**0.0074**
Brain Tumor1	avg	**0.8333**	**0.8333**	**0.8333**	**0.8333**
	std	**0.0000**	**0.0000**	**0.0000**	**0.0000**
Brain Tumor2	avg	**1.0000**	**1.0000**	**1.0000**	**1.0000**
	std	**0.0000**	**0.0000**	**0.0000**	**0.0000**
Breast3	avg	0.5807	0.5807	0.5842	**0.5772**
	std	**0.0096**	**0.0096**	0.0161	**0.0096**
Yale	avg	0.6323	0.6303	0.6293	**0.6212**
	std	0.0132	**0.0123**	0.0130	0.0154
Cancer	avg	0.9858	0.9883	0.9875	**0.9842**
	std	0.0126	0.0127	0.0127	**0.0123**
COIL20	avg	0.9907	0.9898	0.9899	**0.9895**
	std	0.0019	**0.0016**	0.0017	0.0023
DataMatrix	avg	**0.8172**	0.8182	0.8182	0.8182
	std	0.0055	**0.0000**	**0.0000**	**0.0000**
DLBCL	avg	0.9000	0.8933	0.9067	**0.8778**
	std	0.0339	0.0332	0.0332	**0.0253**
DrivFace	avg	0.9752	0.9752	**0.9747**	0.9749
	std	**0.0000**	**0.0000**	0.0021	0.0015
Lung discrete	avg	**1.0000**	**1.0000**	**1.0000**	**1.0000**
	std	**0.0000**	**0.0000**	**0.0000**	**0.0000**
Lymphoma	avg	**0.8947**	**0.8947**	**0.8947**	**0.8947**
	std	**0.0000**	**0.0000**	**0.0000**	**0.0000**

TABLE 16. AVERAGE CALCULATION TIME OF S1-SHAPED TRANSFER FUNCTIONS WITH DIFFERENT ORDER

Datasets	S1(0.4)	S1(0.5)	S1(0.7)	S1(1)
Arrhythmia	8.6974	**8.6398**	8.9945	9.4607
Brain Tumor1	**17.7851**	53.3223	17.8607	19.0214
Brain Tumor2	**21.8602**	23.9264	24.3559	22.7005
Breast3	**16.1885**	16.2435	16.8333	16.9799
Yale	8.5695	**8.5396**	8.7126	8.8468
Cancer	2051.4451	**59.6437**	62.3242	65.0957
COIL20	**99.6747**	102.3630	106.4430	113.8696
DataMatrix	**8.1610**	8.2184	8.3754	8.5933
DLBCL	**15.0457**	15.3636	15.1901	15.8713
DrivFace	**122.8547**	125.2881	129.4459	133.9335
Lung discrete	5.4436	**5.4352**	5.4823	5.4490
Lymphoma	**12.7002**	12.8936	13.1291	12.9594

TABLE 17. AVERAGE NUMBER OF SELECTED FEATURES OF S1-SHAPED TRANSFER FUNCTIONS WITH DIFFERENT ORDER

Datasets	S1(0.4)	S1(0.5)	S1(0.7)	S1(1)
Arrhythmia	**151.6000**	157.5000	159.2667	173.9000
Brain Tumor1	3195.7333	**3158.7667**	3260.5667	3305.5667
Brain Tumor2	5259.4000	**5175.7000**	5352.9000	5485.7333
Breast3	**2590.0000**	2609.1000	2692.4333	2720.4667
Yale	**552.9333**	555.9333	579.1667	603.5667
Cancer	**6658.8000**	6741.7000	6908.9667	7191.3667
COIL20	**566.0333**	570.1000	597.2000	646.7667
DataMatrix	**558.4333**	583.1333	601.0000	637.3333
DLBCL	**2889.9333**	2913.3000	3009.1333	3007.6000
DrivFace	3362.1333	**3353.7333**	3360.0667	3371.8667
Lung discrete	162.5333	164.6667	**160.9667**	162.1333
Lymphoma	2009.7667	2010.6000	**2004.6333**	2019.5000

TABLE 18. AVERAGE AND STANDARD DEVIATION OF ACCURACY S2-SHAPED TRANSFER FUNCTIONS WITH DIFFERENT ORDER

Datasets	Measures	S2(0.4)	S2(0.5)	S2(0.7)	S2(1)
Arrhythmia	avg	0.7293	**0.7281**	0.7285	0.7311
	std	0.0122	**0.0112**	0.0123	0.0125
Brain Tumor1	avg	**0.9444**	**0.9444**	**0.9444**	**0.9444**
	std	**0.0000**	**0.0000**	**0.0000**	**0.0000**
Brain Tumor2	avg	**0.9000**	**0.9000**	**0.9000**	**0.9000**
	std	**0.0000**	**0.0000**	**0.0000**	**0.0000**
Breast3	avg	0.5789	**0.5754**	0.5772	0.5789
	std	**0.0000**	0.0134	0.0096	**0.0000**
Yale	avg	0.6394	**0.6313**	0.6323	**0.6313**
	std	0.0184	0.0161	0.0173	**0.0140**
Cancer	avg	**1.0000**	**1.0000**	**1.0000**	**1.0000**
	std	**0.0000**	**0.0000**	**0.0000**	**0.0000**
COIL20	avg	**0.9826**	0.9833	0.9838	0.9841
	std	0.0026	0.0019	**0.0017**	0.0018
DataMatrix	avg	0.6202	**0.6172**	0.6273	0.6263
	std	0.0191	**0.0149**	0.0254	0.0184
DLBCL	avg	0.9356	0.9356	**0.9333**	0.9356
	std	0.0122	0.0122	**0.0000**	0.0122
DrivFace	avg	0.9356	0.9356	**0.9333**	0.9356
	std	0.0015	0.0000	**0.0000**	0.0000
Lung discrete	avg	**0.8571**	**0.8571**	0.8595	0.8595
	std	**0.0000**	**0.0000**	0.0130	0.0130
Lymphoma	avg	**0.8947**	**0.8947**	**0.8947**	**0.8947**
	std	**0.0000**	**0.0000**	**0.0000**	**0.0000**

TABLE 19. AVERAGE CALCULATION TIME OF S2-SHAPED TRANSFER FUNCTIONS WITH DIFFERENT ORDER

Datasets	S2(0.4)	S2(0.5)	S2(0.7)	S2(1)
Arrhythmia	10.0137	10.3718	9.7721	**9.5397**
Brain Tumor1	25.2235	24.6536	**23.2143**	24.0214
Brain Tumor2	25.0988	48.5432	61.1031	**24.1136**
Breast3	210.0697	**14.2212**	1565.5358	14.5153
Yale	7.9021	7.8911	**7.7968**	7.8436
Cancer	52.0975	50.8768	**50.0268**	208.8021
COIL20	75.9089	73.5495	71.0210	**69.4660**
DataMatrix	7.4080	7.2988	**7.2344**	7.3589
DLBCL	13.8587	13.8058	14.1131	**13.6369**
DrivFace	113.5735	110.2149	103.4522	**98.7530**
Lung discrete	6.9176	5.6304	**5.5763**	5.6732
Lymphoma	13.3435	13.2121	13.0040	**12.5508**

TABLE 20. AVERAGE NUMBER OF SELECTED FEATURES OF S2-SHAPED TRANSFER FUNCTIONS WITH DIFFERENT ORDER

Datasets	S2(0.4)	S2(0.5)	S2(0.7)	S2(1)
Arrhythmia	128.8333	121.1333	118.6000	**113.6667**
Brain Tumor1	2959.0000	**2952.9000**	2963.2333	2958.1000
Brain Tumor2	4879.9667	4755.4000	4516.2000	**4356.4000**
Breast3	2250.5667	2232.2000	2076.8667	**1994.1333**
Yale	460.4667	459.7000	439.5667	**422.3000**
Cancer	6032.5333	5970.7333	5637.4000	**5203.4000**
COIL20	468.2000	458.4333	444.0000	**420.1000**
DataMatrix	471.0333	460.7333	438.4333	**419.9000**
DLBCL	2527.3667	2436.3667	2356.7667	**2200.9667**
DrivFace	2978.7333	2945.6333	2792.8333	**2680.2000**
Lung discrete	155.2000	157.7667	147.8667	**142.3333**
Lymphoma	1979.8333	1999.3000	1978.3000	**1953.4333**

TABLE 21. AVERAGE AND STANDARD DEVIATION OF ACCURACY Z1-SHAPED TRANSFER FUNCTIONS WITH DIFFERENT ORDER

Datasets	Measures	Z1(2)	Z1(5)	Z1(8)	Z1(10)
Arrhythmia	avg	**0.7400**	0.7496	0.7578	0.7593
	std	0.0243	**0.0186**	0.0219	0.0188
Brain Tumor1	avg	**0.8556**	0.8574	0.8574	0.8741
	std	**0.0277**	0.0316	0.0316	0.0324
Brain Tumor2	avg	**0.9367**	0.9433	0.9400	0.9500
	std	**0.0490**	0.0504	0.0498	0.0509
Breast3	avg	**0.8035**	0.8281	0.8193	0.8211
	std	0.0413	0.0364	0.0383	**0.0327**
Yale	avg	**0.6576**	0.6677	0.6717	0.6707
	std	0.0289	**0.0218**	0.0299	0.0261
Cancer	avg	**0.9783**	0.9817	**0.9783**	0.9792
	std	0.0109	0.0112	**0.0086**	0.0095
COIL20	avg	**0.9861**	0.9877	0.9882	0.9889
	std	0.0036	0.0038	0.0038	**0.0032**
DataMatrix	avg	**0.7192**	0.7394	0.7374	0.7293
	std	0.0263	0.0271	0.0291	**0.0251**
DLBCL	avg	**1.0000**	**1.0000**	**1.0000**	**1.0000**
	std	**0.0000**	**0.0000**	**0.0000**	**0.0000**
DrivFace	avg	**0.9755**	0.9782	0.9791	0.9796
	std	0.0046	**0.0041**	0.0042	0.0042
Lung discrete	avg	**0.9571**	0.9619	0.9714	0.9714
	std	**0.0356**	0.0362	**0.0356**	**0.0356**
Lymphoma	avg	**0.8807**	0.8877	0.8930	0.8912
	std	0.0274	0.0182	0.0168	**0.0134**

TABLE 22. AVERAGE CALCULATION TIME OF Z1-SHAPED TRANSFER FUNCTIONS WITH DIFFERENT ORDER

Datasets	Z1(2)	Z1(5)	Z1(8)	Z1(10)
Arrhythmia	**3.8793**	4.5178	4.4190	4.6197
Brain Tumor1	**10.2357**	10.5365	10.4483	10.5867
Brain Tumor2	**15.9258**	16.1955	16.5199	16.4689
Breast3	**9.5146**	9.8580	10.4056	10.4571
Yale	**5.9108**	6.1531	6.1763	6.1751
Cancer	21.2547	22.3052	19.4990	**19.2644**
COIL20	**11.3455**	11.5546	11.7892	12.1022
DataMatrix	**6.1012**	6.2252	6.3255	6.3350
DLBCL	**10.2980**	10.6718	11.0813	11.2504
DrivFace	654.4569	836.7949	**649.3362**	321.5003
Lung discrete	**5.9879**	6.3336	6.3711	6.3564
Lymphoma	9.7485	**9.4507**	15.1543	9.6570

TABLE 23. AVERAGE NUMBER OF SELECTED FEATURES OF Z1-SHAPED TRANSFER FUNCTIONS WITH DIFFERENT ORDER

Datasets	Z1(2)	Z1(5)	Z1(8)	Z1(10)
Arrhythmia	16.6333	12.8667	**9.6667**	13.0000
Brain Tumor1	**29.0000**	42.5333	50.2000	41.8000
Brain Tumor2	**338.2000**	373.9667	553.4000	441.3667
Breast3	**31.5000**	35.1667	92.9333	74.0000
Yale	61.0667	47.2000	**44.3000**	45.3000
Cancer	**934.3333**	1565.9333	1014.2667	1388.4667
COIL20	178.0667	133.9667	106.6333	**103.1667**
DataMatrix	218.9000	183.3667	**159.4667**	176.2333
DLBCL	510.7333	**824.2333**	980.4000	884.7333
DrivFace	51.4667	42.7667	54.6667	**42.4667**
Lung discrete	95.1667	83.7667	72.5333	**66.2667**
Lymphoma	729.6000	401.9667	**373.5667**	406.9667

TABLE 24. AVERAGE AND STANDARD DEVIATION OF ACCURACY Z2-SHAPED TRANSFER FUNCTIONS WITH DIFFERENT ORDER

Datasets	Measures	Z2(0.1)	Z2(0.3)	Z2(0.5)	Z2(0.6)
Arrhythmia	avg	0.7881	0.7900	0.7870	**0.7707**
	std	0.0231	0.0265	0.0226	**0.0205**
Brain Tumor1	avg	0.8833	0.8852	0.8815	**0.8741**
	std	**0.0224**	0.0289	0.0241	0.0250
Brain Tumor2	avg	0.9467	0.8967	0.8833	**0.8467**
	std	0.0571	0.0414	**0.0379**	0.0507
Breast3	avg	0.7596	0.7070	0.6825	**0.6175**
	std	0.0837	0.0766	0.0776	**0.0436**
Yale	avg	0.6889	0.6899	0.6747	**0.6616**
	std	0.0210	**0.0172**	0.0224	0.0196
Cancer	avg	0.9950	0.9900	0.9867	**0.9825**
	std	**0.0102**	0.0125	0.0127	0.0134
COIL20	avg	0.9998	1.0000	1.0000	**0.9997**
	std	0.0009	**0.0000**	**0.0000**	0.0011
DataMatrix	avg	0.7707	0.7727	0.7758	**0.7172**
	std	**0.0294**	0.0295	0.0507	0.0368
DLBCL	avg	**1.0000**	**1.0000**	**1.0000**	**1.0000**
	std	**0.0000**	**0.0000**	**0.0000**	**0.0000**
DrivFace	avg	0.9865	0.9862	0.9810	**0.9780**
	std	0.0063	0.0066	0.0058	**0.0055**
Lung discrete	avg	0.8952	0.8738	0.8714	**0.8619**
	std	0.0449	0.0307	0.0291	**0.0181**
Lymphoma	avg	**0.8947**	0.9000	**0.8947**	0.8982
	std	**0.0000**	0.0161	**0.0000**	0.0134

TABLE 25. AVERAGE CALCULATION TIME OF Z2-SHAPED TRANSFER FUNCTIONS WITH DIFFERENT ORDER

Datasets	Z2(0.1)	Z2(0.3)	Z2(0.5)	Z2(0.6)
Arrhythmia	7.9527	6.9671	6.3489	**6.3135**
Brain Tumor1	16.1238	15.0012	14.2775	**13.7086**
Brain Tumor2	23.1420	21.6340	20.2809	**20.0120**
Breast3	13.7774	12.7404	12.5550	**12.2201**
Yale	8.7485	8.0694	7.4255	**7.1968**
Cancer	52.4622	42.0850	35.7052	**30.4605**
COIL20	98.1444	61.6817	40.5303	**29.0802**
DataMatrix	8.2284	7.6070	7.0778	**7.0205**
DLBCL	16.7148	15.8641	14.4593	**14.0524**
DrivFace	81.5533	61.3869	41.1234	**34.9165**
Lung discrete	5.7831	5.7287	5.8038	**5.6853**
Lymphoma	11.8768	11.7373	11.3391	**11.0751**

TABLE 26. AVERAGE NUMBER OF SELECTED FEATURES OF Z2-SHAPED TRANSFER FUNCTIONS WITH DIFFERENT ORDER

Datasets	Z2(0.1)	Z2(0.3)	Z2(0.5)	Z2(0.6)
Arrhythmia	131.8000	64.4333	25.1667	**16.8333**
Brain Tumor1	1739.5333	1498.2333	1457.3667	**1164.3333**
Brain Tumor2	3822.0333	3451.1000	2282.6667	**1596.1000**
Breast3	926.9000	602.1000	439.2667	**212.5667**
Yale	516.5000	371.2333	222.2667	**164.1000**
Cancer	4425.5667	4030.7000	2995.0667	**2644.8667**
COIL20	537.5333	370.6000	255.0667	**204.6333**
DataMatrix	478.6000	234.8000	114.7667	**75.6333**
DLBCL	2669.4333	2548.9667	2341.6333	**2223.8667**
DrivFace	1303.9667	1111.2333	687.3667	**572.6667**
Lung discrete	107.1333	101.5667	114.2667	**75.3000**
Lymphoma	**872.4667**	1143.6667	1064.2000	937.8000

TABLE 27. AVERAGE AND STANDARD DEVIATION OF ACCURACY OF TRANSFER FUNCTIONS WITH OPTIMAL ORDER

Datasets	Measures	V1（5）	V2(3)	V3(2)	U(10)	S1(0.4)	S2(1)	Z1(2)	Z2(0.6)
Arrhythmia	avg	0.7196	**0.6715**	0.7548	0.7630	0.7022	0.7052	0.7056	0.7663
	std	0.0188	0.0225	0.0137	0.0168	**0.0094**	0.0133	0.0220	0.0158
Brain Tumor1	avg	**0.9444**	**0.9444**	0.9481	0.9500	**0.9444**	**0.9444**	0.9481	0.9463
	std	**0.0000**	**0.0000**	0.0141	0.0170	**0.0000**	**0.0000**	0.0141	0.0101
Brain Tumor2	avg	**0.9567**	**0.9533**	1.0000	1.0000	0.9700	1.0000	1.0000	1.0000
	std	0.0504	0.0507	**0.0000**	**0.0000**	0.0466	**0.0000**	**0.0000**	**0.0000**
Breast3	avg	0.5982	**0.5965**	0.8000	0.7982	0.5789	0.6070	0.7772	0.7579
	std	0.0293	0.0252	0.0487	0.0619	**0.0000**	0.0267	0.0430	0.0612
Yale	avg	0.6404	**0.6283**	0.6889	0.7121	0.6495	0.6576	0.6556	0.7091
	std	0.0154	0.0158	0.0238	0.0284	0.0153	**0.0141**	0.0258	0.0352
Cancer	avg	**1.0000**	**1.0000**	**1.0000**	**1.0000**	**1.0000**	**1.0000**	**1.0000**	**1.0000**
	std	**0.0000**	**0.0000**	**0.0000**	**0.0000**	**0.0000**	**0.0000**	**0.0000**	**0.0000**
COIL20	avg	0.9961	0.9947	0.9969	0.9968	0.9965	0.9965	**0.9921**	0.9966
	std	0.0012	0.0018	0.0014	0.0009	**0.0000**	**0.0000**	0.0020	0.0011
DataMatrix	avg	0.8071	0.7747	0.8192	0.8485	0.8030	0.8051	**0.7657**	0.8434
	std	0.0186	0.0206	0.0232	0.0211	0.0173	**0.0153**	0.0327	0.0240
DLBCL	avg	**1.0000**	**1.0000**	**1.0000**	**1.0000**	**1.0000**	**1.0000**	**1.0000**	**1.0000**
	std	**0.0000**	**0.0000**	**0.0000**	**0.0000**	**0.0000**	**0.0000**	**0.0000**	**0.0000**
DrivFace	avg	0.9893	0.9876	0.9917	0.9917	**0.9835**	0.9837	0.9917	0.9917
	std	0.0039	0.0042	**0.0000**	**0.0000**	**0.0000**	0.0015	**0.0000**	**0.0000**
Lung discrete	avg	**0.8500**	0.8524	0.8952	0.8714	0.8571	0.8571	0.8667	0.8714
	std	0.0218	0.0181	0.0449	0.0291	**0.0000**	**0.0000**	0.0449	0.0346
Lymphoma	avg	0.8386	0.8211	0.8789	0.8702	**0.7930**	0.8018	0.8491	0.8737
	std	0.0237	0.0296	0.0245	0.0301	**0.0134**	0.0226	0.0229	0.0262

TABLE 28. AVERAGE CALCULATION TIME OF 8 KINDS OF TRANSFER FUNCTIONS WITH DIFFERENT ORDERS

Datasets	V1(5)	V2(3)	V3(2)	U(10)	S1(0.4)	S2(1)	Z1(2)	Z2(0.6)
Arrhythmia	9.0589	9.6961	6.0759	6.1637	8.2689	7.5322	**3.9243**	6.0648
Brain Tumor1	19.0380	20.3577	13.3769	14.3012	17.2137	15.9268	**11.9390**	15.1411
Brain Tumor2	21.6088	22.9133	16.1576	16.9008	20.3317	19.2482	**15.4773**	17.1265
Breast3	14.7945	16.7088	10.3248	10.4054	14.7244	13.5379	**8.5143**	10.8777
Yale	8.3428	8.9553	6.1518	6.3418	7.7038	7.2778	**5.6984**	6.4090
Cancer	52.0689	60.9499	27.8807	31.2169	51.9085	43.6910	**19.7554**	33.5309
COIL20	117.1537	138.3974	23.0051	35.3184	87.9767	65.8160	**12.4311**	34.6746
DataMatrix	8.4885	9.0294	6.1967	6.4263	7.6559	7.2244	**5.7974**	6.3912
DLBCL	15.9847	17.2955	11.3050	12.1044	14.5701	13.4919	**9.7614**	12.1888
DrivFace	93.3069	130.7990	22.1837	24.1756	115.7979	95.0033	**13.4231**	33.0218
Lung discrete	5.4727	5.5218	5.2215	5.2895	5.4500	5.4018	**4.1335**	5.2861
Lymphoma	12.7513	13.9385	9.5479	9.7337	12.9947	11.9819	**8.9862**	10.0400

TABLE 29. AVERAGE NUMBER OF SELECTED FEATURES OF 8 KINDS OF TRANSFER FUNCTIONS WITH DIFFERENT ORDERS

Datasets	V1(5)	V2(3)	V3(2)	U(10)	S1(0.4)	S2(1)	Z1(2)	Z2(0.6)
Arrhythmia	208.6667	228.2000	**9.8333**	21.8000	149.9333	112.6333	15.5000	14.8667
Brain Tumor1	3022.3000	3149.9333	2604.2333	**2461.3333**	3000.3000	2922.2667	2596.8667	2747.8667
Brain Tumor2	4807.6000	4656.1333	1656.9000	2210.0000	5492.7000	4247.1000	**780.9000**	2132.6667
Breast3	1910.0333	1990.8333	71.7667	165.0000	2513.4667	2092.5333	**13.5000**	347.8000
Yale	633.9000	629.5000	**93.0667**	128.2333	554.0667	415.6333	105.8667	125.5333
Cancer	6287.7333	6308.6000	**6283.8333**	6290.6333	6307.8000	6306.2667	6305.6667	6303.4667
COIL20	645.4667	665.7000	**227.1333**	286.3667	559.3000	423.6667	422.2000	301.4667
DataMatrix	685.3000	728.2333	**139.2000**	188.3000	568.5000	422.6000	274.3333	156.0000
DLBCL	2937.0667	3364.4333	2034.5333	2068.8000	2848.9667	2430.6667	**1621.1667**	1924.8667
DrivFace	1762.2000	2081.2667	554.7000	486.5000	3230.2000	3063.4333	**259.1667**	672.8667
Lung discrete	133.3333	150.8333	47.4000	72.4000	174.8333	131.0000	**32.8667**	64.5667
Lymphoma	993.8000	1276.0000	273.8333	332.9333	2025.4000	1911.4333	**168.8000**	399.9667

In the first group, the differences of eight groups of transfer functions are simulated in EO and the data are analyzed. In the second group of simulation, the parameters with the best performance in each group of transfer functions are put together for simulation again. Finally, the conclusion that S1(0.4), S2(1), Z1(2) and V2(3) perform better is drawn. It opens up a new way of thinking to solve the feature selection problem with variable order transfer function, and a large number of experimental simulations and data analysis provide strong evidence for the final conclusion. As the future research direction, it is planned to study the dynamic adjustment of transfer function parameters, for example, some new transfer functions related to iteration times or optimization indexes are proposed to enhance the development effect of optimization algorithm.

REFERENCES

[1] L. Huang, J. Yuan and M. Pan. "CFD Simulation and Parameter Optimization of the Internal Flow Field of a Disturbed Air Cyclone Centrifugal Classifier," *Separation and Purification Technology,* vol. 307, pp. 122760, 2023.

[2] K. Gu, F. Xiang and C. Liu. "Insight into the Mechanical Coupling Behavior of Loose Sediment and Embedded Fiber-optic Cable Using Discrete Element Method," *Engineering Geology,* vol. 312, pp. 106948, 2023.

[3] Y. C. Wang, J. S. Wang , J. N. Hou, Y. X. Xing. "Natural Heuristic Algorithms to Solve Feature Selection Problem," *Engineering Letters,* vol. 31, no. 1, pp. 1-18, 2023.

[4] K. Ganesan. "A Novel Deep Learning Approach for Sickle Cell Anemia Detection in Human RBCs Using an Improved Wrapper-based Feature Selection Technique in Microscopic Blood Smear Images," *Biomedical Engineering/Biomedizinische Technik,* vol. 68, no. 2, pp. 175-185, 2023.

[5] M. D. Shah, A. C. Morreale, L. S. Lebel. "Assessment of Filtered Containment Venting Strategies for Mitigating Severe Accident Consequences at a Generic Single-Unit CANDU," *Journal of Nuclear Engineering and Radiation Science,* vol. 9, no. 1, pp.011701, 2023.

[6] M. D. Shah, A. C. Morreale and L. S. Lebel. "Assessment of Filtered Containment Venting Strategies for Mitigating Severe Accident Consequences at a Generic Single-Unit CANDU," *Journal of Nuclear Engineering and Radiation Science,* vol. 9, no. 1, pp. 011701, 2023.

[7] A. J. Kacew, G. W. Strohbehn and L. Saulsberry. "Artificial Intelligence Can Cut Costs While Maintaining Accuracy in Colorectal Cancer Genotyping," *Frontiers in Oncology,* vol. 11, no. 1, pp. 2066, 2021.

[8] T. Luo. "Parallel Genetic Algorithm Based Common Spatial Patterns Selection on Time–frequency Decomposed EEG Signals for Motor Imagery Brain-computer Interface," *Biomedical Signal Processing and Control,* vol.80, pp. 104397, 2023.

[9] W. Zeng, Y. Liao and Y Chen. "Research on Classification and Recognition of the Skin Tumors by Laser Ultrasound Using Support Vector Machine Based on Particle Swarm Optimization," *Optics & Laser Technology,* 158, no. 1, pp. 108810, 2023.

[10] G. Corona, C. O. Maciel and C. J. Morales. "A New Method to Solve Rotated Template Matching Using Metaheuristic Algorithms and the Structural Similarity Index," *Mathematics and Computers in Simulation,* vol. 206, pp. 130-146, 2023.

[11] G. A. Masoud and S. Rashed. "An Exact Criterion Space Search Method for a Bi-objective Nursing Home Location and Allocation Problem," *Mathematics and Computers in Simulation,* vol. 206, pp. 166-180, 2023.

[12] T. V. Ha , Q. H. Nguyen and T. T. Nguyen . "A Parallel Differential Evolution with Cooperative Multi-search Strategy for Sizing Truss Optimization," *Applied Soft Computing,* vol.131, pp. 109762, 2022.

[13] H. Ahmed, M. Hamada and K. Hossam. "Improving the Frequency Response of Hybrid Microgrid under Renewable Sources' Uncertainties Using a Robust LFC-Based African Vulture Optimization Algorithm," *Processes,* vol. 10, no. 11, pp. 2320, 2022.
[14] Kondor G. "NP-hardness of M-dimensional Weighted Matching Problems," *Theoretical Computer Science,* vol. 930, pp. 33-36, 2022.
[15] G. R. Corominas, M. J. Blesa and C. Blum. "AntNetAlign: Ant Colony Optimization for Network Alignment," *Applied Soft Computing,* vol. 132, no. 1, pp. 109832, 2023.
[16] C. Pagliantini, G. Manzini and O. Koshkarov. "Energy-conserving Explicit and Implicit Time Integration Methods for the Multi-dimensional Hermite-DG Discretization of the Vlasov-Maxwell Equations," *Computer Physics Communications,* vol.284, pp. 108604, 2023.
[17] A. A. Saadi, Y. Meraihi and A. Soukane. "A Novel Hybrid Chaotic Aquila Optimization algorithm with Simulated Annealing for Unmanned Aerial Vehicles path planning," *Computers and Electrical Engineering,* vol. 104, pp. 108461, 2022.
[18] D. Yu, Z. Zhang and Y. Deng. "Flexible, Highly Scalable and Cost-effective Network Structures for Data Centers," *Journal of Network and Computer Applications,* vol. 210, pp. 103542, 2023.
[19] Y. S. Khokhlov. "Multivariate Analogs of Classical Univariate Discrete Distributions and their Properties," *Journal of Mathematical Sciences,* vol. 234, pp. 802-809, 2018.
[20] W. A. Witzen, M. L. P. Echlin and M. A. Charpagne. "Subgrain Geometrically Necessary Dislocation Density Mapping in Spalled Ta in Three Dimensions," *Acta Materialia,* vol. 244, pp. 118366, 2023.
[21] M. Zhang, J. S. Wang , J. N. Hou, H. M. Song, X. D. Li and F. J. Guo. "RG-NBEO: A ReliefF Guided Novel Binary Equilibrium Optimizer with Opposition-based S-shaped and V-shaped Transfer Functions for Feature Selection," *Artificial Intelligence Review,* vol. 56, no. 7, pp. 6509-6556, 2023.
[22] M. Qaraad, S. Amjad and N. K. Hussein. "An Innovative Quadratic Interpolation Salp Swarm-based Local Escape Operator for Large-scale Global Optimization Problems and Feature Selection," *Neural Computing and Applications,* vol. 34, no. 20, pp. 17663-17721, 2022.
[23] M. B. Abdel, K. M. Sallam and R. Mohamed. "An Improved Binary Grey-wolf Optimizer with Simulated Annealing for Feature Selection," *IEEE Access,* vol. 9, pp. 139792-139822, 2021.
[24] K. K. Ghosh, P. K. Singh and J. Hong. "Binary Social Mimic Optimization Algorithm with X-shaped Transfer Function for Feature Selection," *IEEE Access,*vol. 8, pp. 97890-97906, 2020.
[25] J. P. Papa, G. H. Rosa and A. N. Souza. "Feature Selection through Binary Brain Storm Optimization," *Computers & Electrical Engineering,* vol. 72, pp. 468-481, 2018.
[26] G. Manita and O. Korbaa. "Binary Political Optimizer for Feature Selection Using Gene Expression Data," *Computational Intelligence and Neuroscience,* vol. 2020, 2020.
[27] E. Pashaei. "Mutation-based Binary Aquila Optimizer for Gene Selection in Cancer Classification," *Computational Biology and Chemistry,* vol. 101, pp. 107767, 2022.
[28] F. O. Jegede, B. A. Adewale and E. J. Ayara. "Users' Perception of Compliance of Security Features with Defensible Space Principles in Selected Secondary Schools in Ota, Ogun State, Nigeria," *IOP Conference Series: Earth and Environmental Science,* vol. 1054, no. 1, pp. 012031, 2022.
[29] N. M. Yusof, A. K. Muda and S. F. Pratama. "Improving Amphetamine-type Stimulants Drug Classification Using Chaotic-Based Time-varying Binary Whale Optimization Algorithm," *Chemometrics and Intelligent Laboratory Systems,* vol. 229, pp. 104635, 2022.
[30] K. Ganesan. "A Novel Deep Learning Approach for Sickle Cell Anemia Detection in Human RBCs Using an Improved Wrapper-based Feature Selection Technique in Microscopic Blood Smear Images," *Biomedical Engineering/Biomedizinische Technik,* vol. 68, no. 2, pp. 175-185, 2023.
[31] K. H. Hui, C. S. Ooi and M. H. Lim. "An Improved Wrapper-based Feature Selection Method for Machinery Fault Diagnosis," *PloS one,* vol. 12, no. 12, pp. e0189143, 2017.
[32] C. T. Tran, M. Zhang and P. Andreae. "Improving Performance for Classification with Incomplete Data Using Wrapper-based Feature Selection," *Evolutionary Intelligence,* vol. 9, pp. 81-94, 2016.
[33] M. Zhang, J. S. Wang, Y. Liu, M. Wang, X. D. Li, F. J. Guo. "Feature Selection Method Based on Stochastic Fractal Henry Gas Solubility Algorithm," *Journal of Intelligent & Fuzzy Systems,* vol. 44, no. 3, pp. 5377-5406, 2023.
[34] I. M. El-Hasnony, S. I. Barakat and M. Elhoseny. "Improved Feature Selection Model for Big Data Analytics," *IEEE Access,* vol. 8, pp. 66989-67004, 2020.
[35] H. W. Chang, Y. H. Chiu and H. Y. Kao. "Comparison of Classification Algorithms with Wrapper-based Feature Selection for Predicting Osteoporosis Outcome Based on Genetic Factors in a Taiwanese Women Population," *International journal of endocrinology,* vol. 2013, 2013.
[36] A. A. Mohammad, A. Vahedi and M. Rahnama. "A Wrapper-based Feature Selection Approach for Accurate Fault Detection of Rotating Diode Rectifiers in Brushless Synchronous Generators," *IOP Conference Series: Materials Science and Engineering,* vol. 671, no. 1, pp. 012045, 2020.
[37] T. Liu, W. Jiao and X. Tian. "Dispatching Optimization of City Gas Station District Energy Systems with Multiple Uncertainties Based on an Improved Affine Arithmetic Method," *Energy Reports,* vol. 9, pp. 37-47, 2023.
[38] Y. Li, W. Wang and J. Liu. "Equilibrium Optimizer with Divided Population Based on Distance and Its Application in Feature Selection Problems," *Knowledge-Based Systems,* vol. 256, pp. 109842, 2022.
[39] Z. Y. Algamal, M. K. Qasim and M. H. Lee. "High-dimensional QSAR/QSPR Classification Modeling Based on Improving Pigeon Optimization Algorithm," *Chemometrics and Intelligent Laboratory Systems,* vol. 206, pp. 104170, 2020.
[40] S. Shen, J. He and X. Wang. "Feature Selection and Online Discrimination for Weak Oxygen Absorption Spectrum," *Optik,* vol. 269, pp. 169917, 2022.