Multiobjective Optimization Algorithm with Dynamic Operator Selection for Feature Selection in High-Dimensional Classification

Wenhong Wei[1], Manlin Xuan[2], Lingjie Li[2]*, Qiuzhen Lin[2], Zhong Ming[2], Carlos A. Coello Coello[3,4]

[1] *School of Computer Science and Technology, Dongguan University of Technology, Dongguan, PR. China*

[2] *College of Computer Science and Software Engineering, Shenzhen University, Shenzhen, PR. China*

[3] *CINVESTAV-IPN, Department of Computer Science, Mexico, D.F. 07360, Mexico*

[4] *School of Engineering and Sciences, Tecnologico de Monterrey, Monterrey, N.L., Mexico*

Abstract:

Feature selection (FS) is an important technique in data preprocessing that aims to reduce the number of features for training while maintaining a high accuracy for classification. In recent studies, FS has been extended to optimize multiple objectives simultaneously in classification. To better solve this problem, this paper proposes a new multiobjective optimization algorithm with dynamic operator selection for feature selection in high-dimensional classification, called FS-DOS. First, two complementary search operators with different characteristics are designed, where the first operator is a quick search (QS) operator aiming to accelerate the convergence speed, and the other operator is a modified binary differential evolution (BDE) operator that can prevent the algorithm from falling into a local optimum. In addition, a dynamic selection strategy based on the idea of resource allocation is also designed to dynamically select the most suitable operator for each solution according to its corresponding performance improvement on aggregated objective values. The simulation results on fifteen different real-world high-dimensional FS datasets show that FS-DOS can obtain a feature subset with higher quality than several state-of-the-art FS algorithms. Importantly, in terms of error rate, FS-DOS wins 55 out of 75 comparisons. In terms of dimensionality reduction, the number of features selected by FS-DOS is between one hundredth and one thousandth of the original dataset.

Keywords: Feature selection, Multiobjective optimization, Evolutionary algorithm, High-dimensional classification, Resource allocation.

1. Introduction

Feature selection (FS) plays an important role in data preprocessing and has been widely used to improve the quality of the feature set in many machine learning tasks, such as classification [1], clustering [2], regression [3], and time series prediction [4]. With the rapid development of the big data era, the dimensions of application data collected from real-world scenes are increasing dramatically, which results in more computational complexity of the FS algorithms for classification. In recent years, the search for the most suitable feature subset from high-dimensional datasets has become a hot research topic in the field of FS. In general, the purpose of FS algorithms is to improve the classification performance by only selecting a small subset of relevant features from the original dataset.

According to the adopted mechanisms, the existing FS algorithms can be roughly classified into three main categories: filter-based FS methods [5]-[6], embedded-based FS methods [7]-[8] and wrapper-based FS methods [9]-[10]. The first kind of FS method uses the filter method to select features, which only considers the inherent characteristics of the data to the dataset [5]. The second class embeds the FS process into learner training in one stage [7], while the last type applies the wrapper method in FS, where the learning performance is used as an evaluation metric for selecting a subset of features [9]. In particular, wrapper-based FS methods require an effective search strategy to guide the evolution of each feature subset; thus, many effective search strategies are embedded into wrapper-based FS methods, which have shown promising performance in classification. For example, the sequential forwards selection (SFS) [11] algorithm initializes the feature sets as the empty set and then selects one feature to be added each time, aiming to find an optimal feature subset. In contrast, the sequential backwards selection (SBS) [12] algorithm starts from the complete feature set and then removes one feature each time until the final optimal feature set is obtained. On the other hand, decision tree [13] searches the best feature subset by dividing the original feature set in turn. Nevertheless, the abovementioned existing FS methods are easily trapped in local optimum when tackling high-dimensional datasets [9].

To address the above issues, many attempts have been made to avoid falling into local optimum by combining some effective techniques with strong global search capabilities [9-10]. Recently, evolutionary algorithms (EAs) have been favored by researchers in the field of FS, especially for solving high-dimensional datasets, such as the genetic algorithm (GA) [14], particle swarm optimization (PSO) [15], and differential evolution (DE) [16], due to their powerful global search capabilities. Note that the above existing works are presented in the form of single-objective EAs that adopt a linear combination method to combine different objectives into a single-objective optimization problem (SOP). However, it is very difficult for linear combination methods to find suitable weight parameters for different objectives. To this end, many studies treat the FS problem as a multiobjective optimization problem (MOP) with two or more conflicting objectives and then apply a multiobjective EA (MOEA) to iteratively search for a set of optimal features, which can achieve a good tradeoff among multiple conflicting objectives. In this way, the decision maker can select an optimal feature subset according to the characteristics of the problem.

Recently, many MOEAs have been presented for solving multiobjective FS problems. For example, Xue *et al*. [15] formulated a multiobjective optimization problem for FS, which aims to optimize the number of selected features and test error rate by using a PSO variant. Nguyen *et al.* [17] used an effective local search strategy to improve the search ability of PSO for evolving feature subsets, which aims to obtain improved classification accuracy. Zhang *et al.* proposed a multiobjective PSO [18] using the ideas of crowding distance, external archiving and Pareto dominance during the PSO search process, which helps to find a subset of features that achieves a high classification accuracy. Nguyen *et al.* [10] proposed a dynamic MOEA based on the decomposition method to solve the high-dimensional FS problem, which can better guide the search in real-world application problems. Xue *et al.* [19] proposed an FS algorithm based on the NSGA-III [20] for dealing with missing dataset classification, which constructs a three-objective optimization problem involving accuracy, feature size and mean imputation [21].

However, the wrapper-based FS algorithms mentioned above usually use a single search strategy, which may be inadequate for various datasets with different preferences, especially for high-dimensional datasets that have many redundant features. To address this issue, this paper proposes a multiobjective optimization algorithm with a dynamic operator selection strategy, called FS-DOS, for high-dimensional classification. First, this paper suggests to using two complementary search operators to improve the scalability and robustness in tackling different high-dimensional datasets. In addition, this paper also investigates an effective mechanism to better utilize these two complementary operators by using the idea of resource allocation (RA), which is widely studied in decomposition-based MOEAs [22]-[24], dynamically selecting the most suitable search operator for different solutions based on their performance improvement. Therefore, the overall performance of FS-DOS is significantly improved. The simulation results show that in terms of the error rate, FS-DOS wins 55 out of 75 comparisons. In terms of dimensionality reduction, the number of features selected by FS-DOS is between one hundredth and one thousandth of the original dataset.

To conclude, the main contributions of our work are clarified as follows:

1) Two complementary search operators with different characteristics are designed for FS in this paper, where the quick search (QS) operator is applied to select the most important features aiming to speed up the convergence, while a modified binary differential evolution (BDE) operator with strong exploration ability is designed to avoid falling into a local optimum.
2) A dynamic operator selection mechanism based on the principle of RA is designed to make better use of the above two search operators, called DOS, which dynamically selects one suitable operator for each solution according to its corresponding performance improvement.
3) The effectiveness of our algorithm is validated on 15 open-source medical datasets with over 2000 dimensions. A large number of simulation results demonstrate that our algorithm has advantages in both accuracy and size of selected features compared to three advanced high-dimensional FS algorithms (i.e., SM-MOEA [25], PS-NSGA [26] and PSOFS-FC [27]) and two traditional FS algorithms (i.e., LFS [28] and CON [29]).

The remaining parts of this paper are organized as follows. Section 2 gives some background knowledge. In addition, some related works on high-dimensional classification are also introduced in Section 2. The details of the proposed algorithm are introduced in Section 3. Next, Section 4 provides the simulation results obtained by FS-DOS and other FS algorithms for performance comparisons. Finally, some conclusions and directions for future works are given in Section 5.

2. Background

This section provides a brief introduction of multiobjective optimization, decomposition approaches, mutual information, DE operators, and some related works on high-dimensional classification.

2.1. Multiobjective optimization

At present, MOEAs are widely used in various fields, such as finance [30] and manufacturing planning [31]. In many machine learning and data mining applications, such as instance selection, pattern mining and community detection [32]-[34], tasks can be treated as multiobjective optimization problems (MOPs). In general, an MOP without any constraint can be defined as follows:

$$\begin{aligned} &\text{minimize} \quad F(x) = (f_1(x), \ldots, f_m(x)) \\ &\text{subject} \quad\;\; x \in \Omega \end{aligned} \tag{1}$$

where $\boldsymbol{x} = (\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n)$ is a decision vector (n is the number of decision variables); $\Omega = [l_i,\ u_i]^n$ is the decision space (l_i and u_i are the lower and upper bounds for the ith variable, $i \in [1,n]$), and $F:\ \Omega \rightarrow R^m$ defines m objective functions (R^m is the objective space).

Given two decision vectors $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$, $\boldsymbol{x}_1$ is said to dominate $\boldsymbol{x}_2$ if and only if $f_i(\boldsymbol{x}_1) \le f_i(\boldsymbol{x}_2)$ for $\forall i \in \{1, \ldots, m\}$ and $f_j(\boldsymbol{x}_1) \ne f_j(\boldsymbol{x}_2)$ for $\exists j \in \{1, \ldots, m\}$, marked by $x_1 \succ x_2$. A solution $\boldsymbol{x}^* \in \Omega$ is regarded as a nondominated solution when no other solution $x \in \Omega$ can dominate x^*. The set of all nondominated solutions is called the Pareto set (PS), and its corresponding set of objective function values is called the Pareto front (PF) [35]. Generally, the objectives are conflicting, and a single solution cannot optimize them simultaneously. The purpose of MOEAs is to find a set of nondominated solutions to evenly approximate the true PF as closely as possible [36]. Particularly, MOEAD [37] proposed by Zhang *et. al.,* is widely used for solving various MOPs, which adopts a number of weight vectors to decompose the target MOP into a set of subproblems and then optimize them separately. There are three commonly used decomposition methods, namely the weighted sum (WS) approach, penalty-based boundary intersection (PBI) approach, and Tchebycheff (TCH) approach [38]-[40].

2.2.Mutual information

Mutual information can be considered as the amount of information in one random variable that is contained in another random variable, defined as follows:

$$I(X;Y) = \sum_{x \in X} \sum_{y \in Y} p(x,y) \log \frac{p(x,y)}{p(x)p(y)} \tag{2}$$

where X and Y are two random variables, $p(x)$ and $p(y)$ refer to the edge probabilities of X and Y, respectively, and $p(x,y)$ means their joint probability density.

Symmetric uncertainty (SU) is obtained by normalizing the mutual information into the entropy of the feature values or the feature values and the label classes [41]. SU is widely applied in many existing FS algorithms [42], [43], which aims to evaluate the correlation between features and label classes, calculated as follows:

$$SU(X,Y) = 2\frac{H(X) - H(X \mid Y)}{H(X) + H(Y)} \tag{3}$$

where $H(X)$ and $H(Y)$ are the entropies of variables X and Y, respectively, and $H(X \mid Y)$ indicates the conditional entropy of X when Y is known. Particularly, $H(X)$, $H(Y)$ and $H(X \mid Y)$ can be defined as follows:

$$H(X) = -\sum_{x \in X} p(x) \log_2 p(x) \tag{4}$$

$$H(Y) = -\sum_{y \in Y} p(y) \log_2 p(y) \tag{5}$$

$$H(X \mid Y) = -\sum_{y \in Y} p(y) \sum_{x \in X} p(x \mid y) \log_2 p(x \mid y) \tag{6}$$

where $p(x)$ and $p(y)$ are the prior probabilities of X and Y, respectively, and $p(x \mid y)$ is the posterior probability of X when Y is given. In this paper, SU is used to evaluate the correlation between each feature and the label.

2.3. Differential evolution operator

When dealing with MOPs, the differential evolution (DE) operator is widely applied during the process of evolutionary search [44], [45] due to its strong global search ability. For each individual (target solution), the DE operator generates a trial solution by (8) and then produces a new solution using the trial solution and target solution by (9), as follows:

$$V_i(t) = X_{r1}(t) + F \times (X_{r2}(t) - X_{r3}(t)) \tag{7}$$

$$V_i(t) = (v_{i,1}(t), v_{i,2}(t), \ldots, v_{i,D}(t))$$
$$v_{i,j}(t) = \begin{cases} u_{i,j}(t), & if\ U(0,1) < CR\ or\ j = j_{rand} \\ x_{i,j}(t), & otherwise \end{cases} \tag{8}$$

where D is the variable dimension, j_{rand} is a random index between [0, D], $X_i(t)$ is the target solution, $U_i(t)$ is the trial solution, $V_i(t)$ is the new solution, and $v_{i,j}(t)$ is an element of $V_i(t)$. Note that $X_{r1}(t)$, $X_{r2}(t)$ and $X_{r3}(t)$ are three solutions randomly selected from the population. The control parameter F is within (0, 2), and CR is within (0, 1). $U(0,1)$ indicates a random value between 0 and 1.

3. Related Work

Over the last decades, many studies have been proposed for high-dimensional classification. For example, Tran *et al.* [9] proposed a new discretized PSO, where a cut-point table is used to store the potential cut points, and the position of the particle is an integer representing the index of the cut point of each feature in the cut-point table. If the cut point does not exist in the cut-point table, it means that the corresponding feature is not selected. Furthermore, Tran *et al.* [46] suggested a variable-length PSO, called VLPSO, which uses the division mechanism to generate particles of different lengths. VLPSO is the first known algorithm to use variable-length PSO for FS, which can help to reduce the search space and improve the performance of the algorithm. Song *et al.* [42] proposed a fast hybrid FS algorithm with three mixed FS strategies. Specifically, the filter method is used to rank the importance of features according to their correlations to labels, and then the clustering method is used to extract similar features. Moreover, the PSO search method is adopted to select important features with low similarity forming the final feature subset. Cheng *et al.* [25] proposed a steering-matrix-based MOEA, where the matrix consisting of the importance of features and the dominance relationship between each of two individuals are designed to guide the evolutionary process. Zhou *et al.* [26] proposed a problem-specific algorithm based on NSGA-III [20] for supervised FS, which takes three objectives (i.e., accuracy, feature size and distance measurement) into consideration. In addition, a fast bit mutation method was designed to randomly flip '0' or '1' for each individual, depending on the number of '0' or '1'. At the same time, a new crossover method was designed, where the children inherit the '1' of the two parents. In this way, a certain number of children in each iteration are generated by using this proposed crossover method. Zhang *et al.* [27] proposed an RF-measure method to evaluate the impact of missing data on the FS performance in the case of unbalanced classes, which applies mutual information to cluster features and then uses a PSO search operator to evolve the population. Tian *et al.* [47] proposed an effective method for dimensionality reduction, which first treats the high-dimensional feature selection problem as a sparse large-scale problem and then adopts two unsupervised neural networks (i.e., a restricted Boltzmann machine and a denoising autoencoder) to quickly reduce the dimensionality of the dataset. A hybrid feature selection algorithm based on a multiobjective algorithm and Relief-F was proposed by Xue *et al.* [48]. The advantages of the filter approach and the wrapper approach are combined to improve the ability to solve FS. First, the features are scored based on their importance to the instance class using the Relief-F algorithm. Then, the feature score information is used to initialize the overall. In addition, a new crossover and mutation operator is designed to guide the crossover and mutation process based on feature scoring information to improve the search direction of the algorithm in the search space and to improve convergence performance. Pan *et al.* [49] proposed an improved grey wolf optimization algorithm for feature selection on high-dimensional data. The algorithm introduces the Relief-F algorithm and coupling entropy in the initialization process, which effectively improves the quality of the initial population. In addition, the improved grey wolf optimization includes two new search strategies. First, a competitive guidance strategy is proposed to update individual positions, making the algorithm's search more flexible. Second, a leader wolf enhancement strategy based on differential evolution is used to find a better position and replace it, thus preventing the algorithm from falling into a local optimum. Li *et al.* [50] proposed an evolutionary multitasking algorithm with a multiple filtering approach for high-dimensional classification. First, generative multitasking using multiple filtering methods is proposed to generate multiple related low-dimensional feature selection tasks. Second, the competitive swarm optimizer is modified to solve the relevant feature selection tasks simultaneously by performing knowledge transfer between them.

Nevertheless, the existing FS algorithms mentioned above only use a single search strategy, which may be inefficient for solving different datasets that have different preferences, especially for solving high-dimensional datasets. To alleviate this problem, this paper proposes two complementary search operators, including a quick search (QS) operator for speeding up convergence and a modified binary deferential evolution (BDE) operator for diversity maintenance. In addition, to better utilize these two search operators, this paper also designs a dynamic operator selection strategy based on the idea of RA, called DOS, to select one suitable search operator for each solution. Our proposed algorithm is described in detail below.

4. The proposed algorithm

At the beginning, the dataset is classified into a training set and a test set, where the training set is considered the original features and fed into the optimization algorithm, while the test set is used to check the performance of the selected features obtained by our method. The main structure of the proposed FS-DOS is plotted in Fig. 1.

The FS problem is first formulated as an MOP with multiple conflicting objectives and then optimized by our proposed FS-DOS method. Note that the balance error rate [51], feature size [26], and distance metric [9] are taken as three conflicting objectives as well as evolution indicators in this paper. Regarding the process of FS-DOS, parameters are initialized in the initialization component. After that, the DOS strategy is executed to select one suitable search operator for each solution. Then, the process of population update is performed. Finally, the performance of the selected features obtained by FS-DOS is evaluated by the test set.

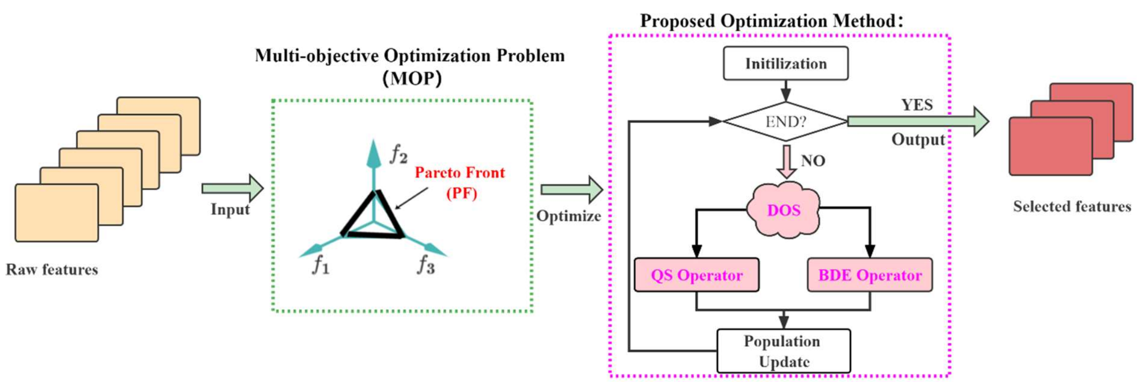


Fig. 1. Structure of our proposed FS-DOS.

4.1. General framework of FS-DOS

Similar to most existing FS algorithms, binary encoding is applied in FS-DOS, where '1' or '0' represents whether the feature is selected or not selected, and the length of each individual in the population equals the number of features in the datasets.

Algorithm 1 Framework of FS-DOS

Input: *maxfes*: maximum number of function evaluations;
N: population size;
Data: the given dataset.
Output: P: population of the last generation.

$F \leftarrow$ set of all features in *Data*;
Initialize the weight vectors $\boldsymbol{\lambda}=\{\lambda_1,\lambda_2,...,\lambda_N\}$, the population P, the ideal point Z^*, the neighbors of each solution $B(i)=\{i_1,i_2,...,i_T\}$;
E, p_{ra}, $p_{repair} \leftarrow$ Initialized by **Algorithm 2**;
for fes=1 **to** $maxfes$ **do**
  for i=1 **to** N **do**
    if rand(0,1)> $p_{ra}(i)$ **then**
      Use **Algorithm 3** to get offspring v_i;
    else
      Use **Algorithm4** to get offspring v_i;
    end if
  Update the ideal point Z^* and the population P using v_i;
  Update the elite set E;
  end for
  Update p_{ra} using (13);
end for

Algorithm 2 Initialization

Input: F: set of all features in *Data*.
Output: E: elite solution set;
p_{ra}: resource allocation probability;
p_{repair}: repair probability.

for i=1 **to** $|F|$
    Calculate SU(F_i) using (3);
    Calculate p_{repair}(F_i) using (10);
end for
for i=1 **to** N
    Calculate $g^{tch}(i)$;
    Initialize $p_{ra}(i)$ =0.35;
end for
$E \leftarrow$ select top 10% solutions with large g^{tch};

The pseudocode of the general framework of the proposed FS-DOS is given in Algorithm 1, where *Data* are the given dataset and *maxfes* and N are the maximum number of function evaluations and population size, respectively. As shown in Algorithm 1, the proposed FS-DOS consists of three main components. First, the initialization process is performed in lines 1 to 3, where the population is initialized and the neighbors of each solution are determined. In addition, the repair probability (i.e., p_{repair}) corresponding to each feature, the initial resource allocation probability (i.e., p_{ra}) and the initial elite solution set E are initialized by Algorithm 2. Then, the process of our proposed FS-DOS is performed in lines 4 to 15. As shown in line 4, the designed RA-based DOS strategy is performed to select the QS operator or BDE operator for each individual based on its corresponding improvement on the aggregated objective value. The details of these two search operators are introduced in Algorithm 3 and Algorithm 4, respectively. The ideal point Z^*, the current population P and elite set E are sequentially updated in lines 11-12. After that, the RA-based probability p_{ra} is updated by using Equation (13). Finally, the population P is output when the termination condition is satisfied.

4.2. Two complementary search operators

Compared with low-dimensional datasets, high-dimensional datasets often involve many redundant features; thus, it is very difficult to find a suitable subset of features in a high-dimensional dataset. To address this issue, FS-DOS proposes two complementary search operators during the evolutionary process, including a QS operator and a BDE operator. On the one hand, the BDE operator is considered an improved variant of the DE operator, and some related literature listed below demonstrates the strong exploration ability of the DE operator [44] [45]. On the other hand, the QS operator is a new operator specifically set up for high-dimensional feature selection datasets, aiming to quickly eliminate redundant features. The details of each search operator are introduced as follows:

4.2.1. Quick search operator

To tackle high-dimensional datasets more effectively, a QS operator is designed in FS-DOS, including an elimination strategy and a repair strategy. The pseudocode of the proposed QS operator is given in Algorithm 3.

First, as shown in lines 1-2, two solutions are randomly selected from the neighbors of the current solution to execute the basic uniform crossover. Then, the elimination probability p_e in the QS operator is calculated in line 3, which is dynamically adjusted based on the generation counter t, formulated as follows:

$$p_e = (1 - p_{\min}) \times \frac{1}{(1 + p_{\min} \times \exp(10 \times (\frac{t}{T} - 0.6)))} \tag{9}$$

where $p_{\min}$ indicates the minimum elimination probability and t and T represent the current generation and the maximum generation, respectively. To intuitively observe the changing tendency of p_e based on the current iteration t, the graph of p_e changing with t is plotted in Fig. 2. As we can learn from that, the value of p_e is very large at the early evolutionary stage, while it decreases slowly with increasing t. Thus, a strong elimination ability is obtained to remove many redundant features at the early evolutionary stage, which is crucial for solving a high-dimensional dataset, as it can accelerate the convergence speed. On the other hand, at the later evolutionary stage, only a gentle elimination ability is needed, as more attention should be focused on the accuracy.

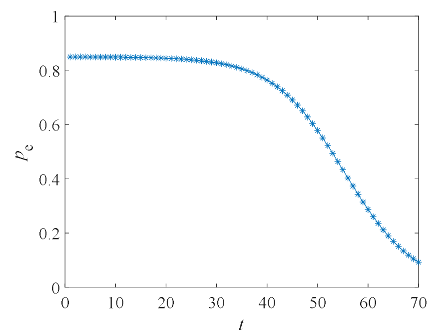


Fig. 2. The graph of p_e changing with iteration t.

Second, to protect some valuable features simultaneously, a repair strategy by inheriting the elite solution is also designed in the proposed QS operator, aiming to cooperate with the eliminate strategy, formulated as follows:

$$p_{repair}(i) = \frac{SU(x_i, y)}{\max\limits_{j=1,2,\ldots,D} SU(x_j, y)} \quad i = 1, 2, \ldots, D \tag{10}$$

where $SU(x_i, y)$ indicates the correlation between the feature x_i and label y, calculated in (4), and D is the number of features in the original dataset. In the repair phase, we first select an elite solution e_{index} from the elite solution set E in line 10. Then, as shown in lines 11-17, if and only if the aggregated objective values of the eliminated solution x_i are larger than those of e_{index}, the eliminated solution x_i is considered to be repaired. Specifically, the current solution is repaired by the feature bit of the elite solution e_{index}, which is '1' based on the repair probability p_{repair}.

Algorithm 3 Quick search operator

Input: p_{repair}: repair rate;
p_e: elimination rate;
F: set of all features in data;
E: elite solution set;
$B(i)$: neighbor of the current solution;
t: current generation;
T: max generation.
Output: v_i: offspring solution.
$X_{r1}, X_{r2} \leftarrow$ randomly selected from $B(i)$;
$x_i = uniform\_crossover\ (X_{r1}, X_{r2})$;
//Elimination strategy
Calculate p_e using (10);
$v_i = x_i$;
for j=1 **to** $|F|$
  if p_e >rand **then**
    $v_{i,j}$ =0;
  end if
end for
//Repair strategy
$e_{index} \leftarrow$ randomly choose a solution from E;
if $g^{tch}(e_{index} \mid \boldsymbol{\lambda_{index}}, z^*) < g^{tch}(v_i \mid \boldsymbol{\lambda_i}, z^*)$
  for j=1 **to** $|F|$
    If $e_{index,j}$ =1 **and** $p_{repaire}(j)$>rand **then**
      $v_{i,j}=e_{index,j}$;
    end if
  end for
end if

4.2.2. Modified binary differential evolution

Zhang *et al.* [45] proposed a binary differential evolution operator (BDE) to address FS problem, showing a strong global search ability for binary coding problems, but it is not suitable for high-dimensional FS. Thus, a novel modified BDE operator is designed in FS-DOS for tackling high-dimensional classification, aiming to prevent QS operation from falling into a local optimum. The pseudocode of the proposed BDE operator is given in Algorithm 4, and three solutions X_{r1}, X_{r2}, X_{r3} are first selected from the neighbors $B(i)$ of the current individual. Then, an optimal solution X_{best} is determined by comparing the nondominant relations among these three selected individuals, and then the flipping probability p_{flip} of each feature is calculated as follows:

$$p_{flip} = F \times rand \times (X_1(t) \oplus X_2(t)) + \sigma \tag{11}$$

where X_1, X_2 are two solutions in X_{r1}, X_{r2}, X_{r3} other than X_{best} and σ is a threshold coefficient to ensure that each feature has the possibility of being flipped. A new individual u_i is obtained through the above steps. Then, u_i and x_i are crossed by the ordinary DE operation through (8) to obtain the offspring v_i. A schematic of our proposed BDE operator is shown in Fig. 3.

Algorithm 4 Binary differential evolution operator

Input: F: set of all features in data;
E: elite solution set;
X_i : current solution;
$B(i)$: neighbor of the current solution;
t: current generation;
T: maximal generation.
Output: v_i: offspring solution.
$X_{r1}, X_{r2}, X_{r3} \leftarrow$ randomly selected from $B(i)$;
$X_{best} \leftarrow$ best solution in X_{r1}, X_{r2}, X_{r3};
$X_1, X_2 \leftarrow$ two solutions in X_{r1}, X_{r2}, X_{r3} other than X_{best};
Calculate p_{flip} by (11);
$u_i = X_{best}$;
for j=1 **to** $|F|$
    if $p_{flip}(j)$ > rand(0,1) **then**
        $u_{i,j}$ =1- $u_{i,j}$;
    end if
end for
get v_i by (8);

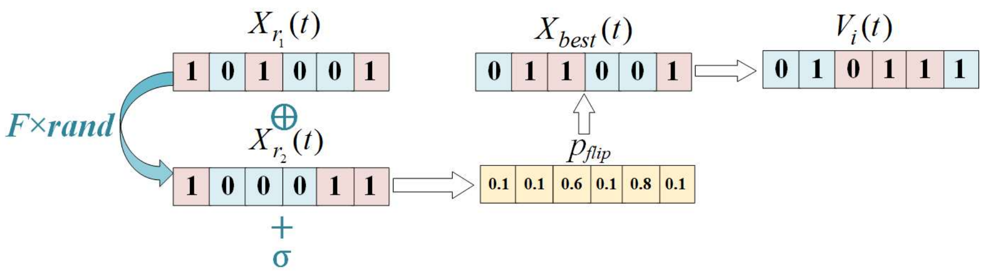


Fig. 3. A schematic of the proposed BDE operator.

In summary, our proposed QS operator has the advantage of rapidly reducing the feature dimensions and eliminating a large number of redundant features, but it easily falls into a local optimum. On the other hand, the BDE operator is designed with strong exploration ability that helps to jump out of the local optimum. Thus, the collaboration of the two complementary search operators is an effective way to improve the scalability and robustness of our method for classification in high-dimensional datasets.

4.3. Dynamic operator selection based on resource allocation

The main purpose of the RA strategy in MOEAs [31-33] is to assign different computational resources for different objectives (i.e., individuals, operators, subregions, etc.) based on the corresponding performance. Following the idea of the RA strategy, this paper designs an RA-based dynamic operator selection strategy, called DOS, to dynamically select the most suitable operator for each solution according to the performance improvement on aggregated objective values at every iteration, calculated as follows:

$$\Delta^i = \frac{g^{tch}(x^i_{t-\Delta t} \mid \lambda^i, z^*) - g^{tch}(x^i_t \mid \lambda^i, z^*)}{g^{tch}(x^i_{t-\Delta t} \mid \lambda, z^*)} \tag{12}$$

where t is the current generation, Δt is the update period, and $g^{tch}(\cdot)$ means the aggregation objective value constructed by using the TCH approach. Fig. 4 shows a schematic diagram of the improvement in the aggregated objective value of each subproblem (i.e., Δ^i). The RA-based selection probability p_{ra} for each solution is based on its corresponding Δ^i, calculated as follows:

$$p_{ra} = \alpha \times \frac{\Delta^i + \varepsilon}{\max\limits_{j=1,2,\ldots,N} \{\Delta^j\} + \varepsilon} + \beta \tag{13}$$

where $\varepsilon = 10^{-5}$ is a small value to guarantee a valid division and α and β are two control parameters, which are set as 0.3 and 0.2, respectively.

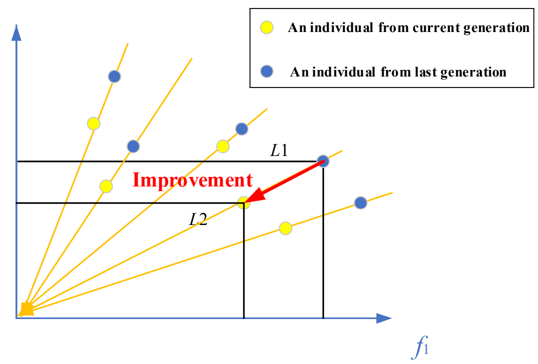


Fig. 4. An example of the performance improvement on the aggregated objective value of each subproblem (L_1 and L_2 are TCH contour lines).

Regarding the principle of our DOS strategy, a higher performance improvement on aggregated objective values indicates a better evolutionary effect of this solution in the previous iterations. In other words, when the performance improvement of a solution is high, this solution is more likely to choose the BDE operator for enhancing the diversity of the population. In contrast, if its performance improvement is low, it indicates that this solution prefers the QS operator to speed up convergence. In this way, these two complementary search operators can cooperate well by using the proposed DOS strategy, which can effectively improve the robustness and scalability of the FS algorithm in high-dimensional classification.

4.4. Fitness function

The fitness function of FS-DOS is set with reference to PS-NSGA [26]. The balance error rate, feature size and distance metric are used as the three objectives. First, since the data imbalance phenomenon exists in most high-dimensional datasets, the balanced classification error was chosen as the first indicator, which can be calculated as follows:

$$\begin{aligned} balanced\_error &= 1 - balanced\_accuracy \\ balanced\_accuracy &= \frac{1}{c}\sum_{i=1}^{c} TPR_i \end{aligned} \tag{14}$$

where c is the number of classes in the dataset and TPR_i is the true positive ratio or the proportion of correctly predicted samples in class i. All classes are treated equally with the weight of $1/c$.

Second, the feature size is used as the second indicator, which requires obtaining a smaller feature subset. The feature size can be calculated as follows:

$$feature\_size = \frac{number\ of\ selected\ features}{D} \tag{15}$$

where D is the number of features in the original dataset.

Third, the classifier is easy to overfit in some cases, which may cause unreliable accuracy of the training set. The distance metric is used to maximize the distance between instances of different classes (D_B) and minimize the distance between instances of the same class (D_W). The distance metric can be calculated as follows:

$$\text{distance} = \frac{1}{1+\exp^{-5(D_B - D_W)}} \tag{16}$$

$$D_B = \frac{1}{|N|}\sum_{i=1}^{|S|} \min_{\{j|j\neq i, class(S_i)\neq class(S_j)\}} Dis(S_i, S_j) \tag{17}$$

$$D_W = \frac{1}{|N|}\sum_{i=1}^{|S|} \max_{\{j|j\neq i, class(S_i)= class(S_j)\}} Dis(S_i, S_j) \tag{18}$$

where $|N|$ is the number of samples in the training data and $Dis(V_i, V_j)$ is the distance between two samples S_i and S_j. Please note that the Manhattan distance is used to measure the distance between them.

5. Simulation results and discussion

In this section, FS-DOS is compared with three state-of-the-art FS algorithms and two traditional FS algorithms on 15 high-dimensional datasets. In addition, ablation simulations are also performed to verify the effectiveness of our proposed evolutionary operators and the proposed DOS strategy. Note that all the simulations are run by using MATLAB R2020a on a 64-bit Windows 10 personal computer with 24 GB RAM and an Intel Core-i7 3.6 GHz processor.

5.1. Simulation settings

5.1.1. Datasets

Fifteen gene-expression high-dimensional datasets [9], [44] are adopted in this paper to evaluate the performance of our proposed algorithm. Table 1 describes the details of these adopted datasets, where the datasets are sorted in ascending order of the feature dimensions. Specifically, the ‘SRBCT’ dataset has the lowest dimension, with only 2,308 features, while the ‘Lung Cancer’ dataset has the highest dimension, with 12,600 features. Note that we have deposited the source codes in the Code Ocean platform, which is available at https://codeocean.com/capsule/e87f078e-d923-4eeb-9d72-1394f6d4c727/.

To ensure that the comparison results are fair and effective, the tenfold cross validation (CV) method is adopted in our simulations to estimate the results. Specifically, each dataset D is first divided into ten groups, nine of which are used as training sets, while the remaining one is used as the test set. Moreover, during the evolutionary process, the tenfold CV method is also used for the training set in each iteration, aiming to evaluate the objective values of current solutions. Note that the number of categories in the test set should be proportional to the number of categories in each partition’s training set.

Table 1 Summary of Datasets

No.	Datasets	#Features	#Instances	#Classes
1	SRBCT	2,308	83	4
2	Lymphoma	5,026	62	3
3	Breast3	4,869	95	3
4	Nci	5,244	61	8
5	Leukemia 1	5,327	72	3
6	DLBCL	5,469	77	2
7	9 Tumor	5,726	60	9
8	Brain Tumor 1	5,920	90	5
9	Prostate6033	6,033	102	2
10	Adenocarcinoma	9,868	76	2
11	Brain Tumor 2	10,367	50	4
12	Prostate	10,509	102	2
13	Leukemia 2	11,225	72	3
14	11 Tumor	12,533	174	11
15	Lung Cancer	12,600	203	5

5.1.2. Compared algorithms

To verify the effectiveness of the proposed FS-DOS, SM-MOEA [25], PS-NSGA [26], PSOFS-FC [27], linear forwards selection (LFS) [28] and consistency-based search (CON) [29] are selected as the compared methods. Specifically, SM-MOEA, PS-NSGA and PSOFS-FC are the three state-of-the-art FS methods, and LFS and CON are two traditional two-stage FS methods that apply the MDL method [52] to prehandle datasets before FS. In addition, two baseline competitors, namely, “FULL” and MOEA/D-FS, are also adopted in our simulations for performance comparisons. Specifically, “FULL” means that KNN with k =1 is directly used to classify datasets without FS, while MOEA/D-FS only uses binary coding, simple crossover and mutation operations based on the MOEA/D framework.

5.1.3. Parameter settings

In terms of parameter settings, we use weka with the default parameters [53] for two traditional FS algorithms, i.e., CON and LFS. Some parameter settings for other compared FS algorithms are summarized in Table 2. Note that both wrapper methods and hybrid methods use the performance of the classifier as the evaluation standard of the feature subset during the evolutionary process. To be fair, we adopt KNN (k=1) as the classifier for all compared algorithms, and the maximum number of iterations and population size for all algorithms are set to 100 and 300, respectively. Note that there are both single-objective and multiobjective optimization algorithms for performance comparisons, while the error rate is a key concern in most FS algorithms. Thus, we choose the individual with the lowest training error rate from the final solution set at the last iteration.

The simulation results obtained by all the compared algorithms are summarized in Table 3 and Table 5. Note that the sizes and error rates in the tables represent the average number of selected features and average error rates, respectively, which are obtained by each compared algorithm after 30 independent runs. To provide an intuitive observation, the lowest size and error rate obtained for each dataset are indicated in bold. In addition, Column *S* represents the Wilcoxon significance test with a significance of 0.05, where '+', '-' and '≈' indicate that the algorithm is better than, worse than and similar to our proposed FS-DOS in terms of the error rate.

Table 2 Parameter settings

Algorithm	Parameters
SM-MOEA	Attenuation factor γ=0.1
PS-NSGA	Mutation probability=0.1, Mutation retry number=1
PSOFS-FC	c1=c2=2, ω=1.0123, θ=0.6, α=k/WM_rate
FS-DOS	p_{min}=0.15, CR=0.4, F=0.5, α=0.3, β=0.2

5.2.Comparisons between FS-DOS and state-of-the-art evolutionary-based FS methods

Research Question 1: Is our proposed algorithm superior in terms of error rate and number of features selected compared to the latest EA-based feature selection algorithms?

A comparison with FULL provides the most intuitive view of the advantages of FS-DOS. From Table 3, FS-DOS only selects approximately one-thousandth of the features in most of the original datasets, while its error rates for classification are greatly reduced. Specifically, in the '9 Tumor' dataset, FS-DOS only selects 1.8% of the original features, while its error rate is decreased by 16.83% when compared with the FULL using all features. In the 'Brain Turom2' dataset, FS-DOS has reduced the feature dimension to 0.4% of the original features, and its error rate is decreased by 10%. In the 'Prostrate' dataset, FS-DOS reduces the feature dimension by 99.92% for the original features, while its error rates also show obvious advantages when compared to the FULL. MOEA/D-FS is the classical MOEA, and compared to MOEA/D-FS, it can better represent the advantages of FS-DOS in dealing with high-dimensional datasets. As shown in Table 3, the number of features selected by MOEA/D-FS is 100 to 1000 times larger than that of our proposed FS-DOS. Furthermore, the error rate obtained by FS-DOS is significantly lower than that of MOEA/D-FS in all adopted datasets except the 'Lymphoma' dataset. Note that with increasing dimensionality, MOEA/D-FS shows poorer performance in dimensionality reduction and error rate. Particularly, in the '9 Tumor' dataset, the number of features selected by MOEA/D-FS is 40 times larger than that of our proposed FS-DOS, and its error rate is approximately 10% higher than that of FS-DOS. In the 'Brain Tumor2' dataset, the number of selected features in MOEA/D-FS is 100 times larger than that of FS-DOS, while its error rate is 8% higher than that of FS-DOS.

The fitness function chosen for PS-NSGA is the same as that chosen for FS-DOS. Compared with PS-NSGA, the advantages of FS-DOS in terms of operator design can be better reflected. As shown in Table 3, the dimensionality reduction effect of FS-DOS is significantly better than that of PS-NSGA in most adopted datasets. Furthermore, the error rate obtained by FS-DOS is also much lower than that of PS-NSGA in 9 out of all 15 datasets. Notably, although the error rate of FS-DOS is similar to that of PS-NSGA in the '9 Tumor', 'Leukemia2' and '11 Tumor' datasets, the number of features selected by our proposed FS-DOS is much smaller than that of PS-NSGA. Therefore, we can conclude that FS-DOS has obvious advantages in dimensionality reduction and error rates compared with PS-NSGA.

Table 3 Average test results of FS-DOS versus competitors

Dataset	Algorithm	Size	Error	S	Dataset	Algorithm	Size	Error	S
SRBCT	FULL	2038	12.92	-	Prostate6033	FULL	6033	18.69	-
	SM-MOEA	14.97	12.72	-		SM-MOEA	**9.89**	18.28	-
	PS-NSGA	**4.93**	10.24	-		PS-NSGA	46.52	16.03	-
	PSOFS-FC	67.6	9.33	-		PSOFS-FC	56.4	18.67	-
	MOEA/D-FS	1080.4	4.08	≈		MOEA/D-FS	2857.3	19.67	-
	FS-DOS	13.47	**3.82**			FS-DOS	24.23	**13.75**	
Lymphoma	FULL	4026	**0.82**	+	Adenocarcinoma	FULL	9868	37.26	-
	SM-MOEA	71	23.59	-		SM-MOEA	**11.49**	**30.40**	+
	PS-NSGA	**2.1**	7.99	≈		PS-NSGA	55.95	36.06	-
	PSOFS-FC	9.8	63.33	-		PSOFS-FC	16.8	50.00	-
	MOEA/D-FS	1896.7	2.2	+		MOEA/D-FS	4660.5	40.48	-
	FS-DOS	3.2	7.33			FS-DOS	28.26	35.38	
Breast3	FULL	1536	44.77	-	Brain Tumor 2	FULL	10367	37.50	-
	SM-MOEA	17.5	**29.01**	+		SM-MOEA	**9.27**	42.79	-
	PS-NSGA	127	44.72	-		PS-NSGA	74.66	33.67	-
	PSOFS-FC	**13.2**	58.70	-		PSOFS-FC	28.6	37.50	-
	MOEA/D-FS	2315.5	43.44	-		MOEA/D-FS	4987.8	35.00	-
	FS-DOS	113.3	40.59			FS-DOS	43.2	**27.50**	
Nci	FULL	5244	32.74	-	Prostate	FULL	10509	14.67	-
	SM-MOEA	15.06	34.51	-		SM-MOEA	13.5	17.19	-
	PS-NSGA	125.23	32.83	-		PS-NSGA	63.2	**12.15**	+
	PSOFS-FC	**13.8**	58.75	-		PSOFS-FC	77.8	26.83	-
	MOEA/D-FS	2495.8	31.42	≈		MOEA/D-FS	5065.88	16.72	-
	FS-DOS	179.77	**29.60**			FS-DOS	**8.2**	13.62	
Leukemia 1	FULL	5327	20.28	-	Leukemia2	FULL	11225	10.56	≈
	SM-MOEA	**9.16**	22.12	-		SM-MOEA	**8.56**	16.86	-
	PS-NSGA	16.8	**12.89**	+		PS-NSGA	28.06	10.94	≈
	PSOFS-FC	13.8	54.33	-		PSOFS-FC	13.4	22.33	-
	MOEA/D-FS	2535.4	17.31	≈		MOEA/D-FS	5420.7	12.40	-
	FS-DOS	29.2	16.67			FS-DOS	14.5	**9.44**	
DLBCL	FULL	5469	17.00	-	11 Tumor	FULL	12533	28.58	-
	SM-MOEA	12.8	19.63	-		SM-MOEA	46.44	29.71	-
	PS-NSGA	**11.2**	15.95	-		PS-NSGA	334.72	22.19	≈
	PSOFS-FC	106.8	**8.50**	+		PSOFS-FC	**15.4**	41.11	-
	MOEA/D-FS	2629.6	15.83	-		MOEA/D-FS	6096.3	25.62	-
	FS-DOS	13.7	10.00			FS-DOS	94.1	**21.36**	
9 Tumor	FULL	5726	63.33	-	Lung Cancer	FULL	12600	21.95	-
	SM-MOEA	14.53	52.64	-		SM-MOEA	20.69	**12.94**	+
	PS-NSGA	192.2	**45.85**	≈		PS-NSGA	106.3	13.50	+
	PSOFS-FC	**14.4**	72.08	-		PSOFS-FC	**19**	74.00	-
	MOEA/D-FS	2753.9	56.17	-		MOEA/D-FS	6027.6	21.17	≈
	FS-DOS	98.9	46.50			FS-DOS	33.8	20.37	
Brain Tumor 1	FULL	5920	27.92	-					
	SM-MOEA	**6.4**	25.65	-					
	PS-NSGA	61.11	28.76	-					
	PSOFS-FC	27.2	62.83	-					
	MOEA/D-FS	2823	24.84	≈					
	FS-DOS	55.5	**23.57**						

Although both SM-MOEA and PSOFS-FC show a great advantage in dimensionality reduction, they may fall into a local optimum in terms of error rate, with FS-DOS having a greater advantage in terms of error rate. Although SM-MOEA shows the advantage in dimensionality reduction, the error rates obtained by FS-DOS are much smaller than those of SM-MOEA in most adopted datasets. Specifically, the number of features selected by our proposed FS-DOS is on the same order of magnitude as that of SM-MOEA. Conversely, the error rate of FS-DOS shows obvious advantages in 12 out of all 15 adopted datasets, while FS-DOS is only inferior to SM-MOEA in 3 datasets, i.e., 'Breast3', 'Adenocarcinoma' and 'Lung Cancer' datasets. Compared to PAOFS-FC among these 15 adopted datasets, the dimensionality reduction effect of PSOFS-FC is only better than our proposed FS-DOS in two datasets (i.e., 'Breast3' and 'Nci'), while the dimensionality reduction effect of PSOFS-FC on other datasets maintains the same level as that of our proposed FS-DOS. However, the error rate of FS-DOS is much lower than that of PSOFS-FC in all adopted datasets except the 'DLBCL' dataset. Therefore, compared with PSOFS-FS, our proposed FS-DOS shows advantages in error rate and similar performance in dimensionality reduction effect in most adopted datasets.

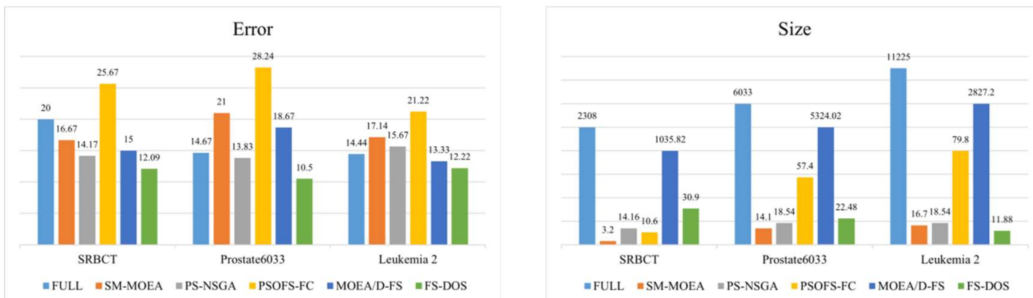
In summary, in a total of 75 comparisons across 15 datasets used, FS-DOS wins 55 times, draws 11 times, and only loses 9 times. Among them, FS-DOS ranks first in 7 out of 15 datasets, while PS-NSGA, SM-MOEA, *FULL* and PSOFS-FC rank first in 3, 3, 1 and 1 datasets, respectively. Therefore, the simulation results listed in Table 3 validate that our proposed FS-DOS is effective for processing high-dimensional datasets. The superiority of FS-DOS over other advanced FS algorithms is mainly due to the combination of these two complementary search operators by using the DOS strategy.

In addition, to further validate the effectiveness of the proposed algorithm, a nonparametric effect size comparison using Cliff's delta [54] between FS-DOS and other algorithms is given in Table 4. Cliff's delta represents the overlap between the two groups of samples, and it ranges from -1 to 1. In general, when $|\boldsymbol{\delta}|$ is less than 0.147, the two groups of samples are close to each other. When $|\boldsymbol{\delta}|$ is between 0.147 and 0.33, these two groups of samples are slightly different. When $|\boldsymbol{\delta}|$ is between 0.33 and 0.474, these two groups of samples are moderately different. When $|\boldsymbol{\delta}|$ is greater than 0.474, there is a large difference between the two groups of samples. As seen from the results listed in Table 4, FS-DOS has a medium or large difference from the state-of-the-art algorithm on most datasets.

Table 4 nonparametric effect size comparison using Cliff's delta

Dataset	Algorithm	$\lvert\delta\rvert$	Dataset	Algorithm	$\lvert\delta\rvert$	Dataset	Algorithm	$\lvert\delta\rvert$
SRBCT	FULL	0.55	DLBCL	FULL	0.66	Brain Tumor 2	FULL	0.76
	SM-MOEA	0.52		SM-MOEA	0.71		SM-MOEA	0.55
	PS-NSGA	0.49		PS-NSGA	0.64		PS-NSGA	0.44
	PSOFS-FC	0.47		PSOFS-FC	0.47		PSOFS-FC	0.59
	MOEA/D-FS	0.14		MOEA/D-FS	0.59		MOEA/D-FS	0.46
Lymphoma	FULL	0.76	9 Tumor	FULL	0.86	Prostate	FULL	0.40
	SM-MOEA	1.00		SM-MOEA	0.47		SM-MOEA	0.52
	PS-NSGA	0.11		PS-NSGA	0.18		PS-NSGA	0.39
	PSOFS-FC	1.00		PSOFS-FC	1.00		PSOFS-FC	0.47
	MOEA/D-FS	0.53		MOEA/D-FS	0.53		MOEA/D-FS	0.33
Breast3	FULL	0.34	Brain Tumor 1	FULL	0.60	Leukemia2	FULL	0.13
	SM-MOEA	1.00		SM-MOEA	0.34		SM-MOEA	0.36
	PS-NSGA	0.28		PS-NSGA	0.37		PS-NSGA	0.10
	PSOFS-FC	0.96		PSOFS-FC	1.00		PSOFS-FC	0.56
	MOEA/D-FS	0.33		MOEA/D-FS	0.16		MOEA/D-FS	0.34
Nci	FULL	0.32	Prostate6033	FULL	0.23	11 Tumor	FULL	0.60
	SM-MOEA	0.52		SM-MOEA	0.26		SM-MOEA	0.71
	PS-NSGA	0.36		PS-NSGA	0.21		PS-NSGA	0.19
	PSOFS-FC	1.00		PSOFS-FC	0.26		PSOFS-FC	1.00
	MOEA/D-FS	0.20		MOEA/D-FS	0.35		MOEA/D-FS	0.20
Leukemia 1	FULL	0.45	Adenocarcinoma	FULL	0.60	Lung Cancer	FULL	0.84
	SM-MOEA	0.53		SM-MOEA	0.41		SM-MOEA	0.66
	PS-NSGA	0.36		PS-NSGA	0.36		PS-NSGA	0.42
	PSOFS-FC	1.00		PSOFS-FC	0.78		PSOFS-FC	1.00
	MOEA/D-FS	0.20		MOEA/D-FS	0.53		MOEA/D-FS	0.13

Furthermore, to study the performance of all compared algorithms on different classifiers, we also adopt another classifier in our experiments, i.e., the decision tree. Without loss of generality, we compare the performance of all comparison algorithms on three representative datasets, including the 'DLBCL' dataset with a small number of features, the 'Prostate6033' dataset with a medium number of features, and the 'Leukemia 2' dataset with a large number of features. Fig.5 provides the experimental results of all the compared algorithms on these three adopted datasets.



(a) (b)

Fig.5. Experimental results of all comparison algorithm on the decision tree classifier: (a) error rate, (b) feature size.

As seen from the figure, FS-DOS still has good performance in terms of both the error rate and the feature size on the decision tree classifier. Specifically, FS-DOS achieves the lowest error rate on these three datasets. In particular, FS-DOS achieves not only the lowest error rate but also the smallest subset of features on the 'Leukemia 2' dataset, which includes a large number of features. Therefore, we can conclude that FS-DOS performs better as the dimensionality of the dataset increases. In summary, the experimental results on the decision tree classifier further validate the effectiveness of our method compared with other methods.

5.3.Comparisons between FS-DOS and no evolution-based FS methods

Research Question 2: Although the last simulation proved that the FS-DOS is effective compared to the latest feature selection algorithms, how does it compare to some of the traditional non-EA algorithms?

Table 5 provides the comparison results of two traditional methods, including MDL+LFS [28] and MDL+CON [29], and our proposed FS-DOS in 15 high-dimensional datasets.

Compared with the two traditional methods, FS-DOS has no advantage in terms of the number of selected features, while it still maintains one and two orders of magnitude. More importantly, our proposed FS-DOS significantly outperforms these two traditional algorithms in terms of error rates. Specifically, the error rate of FS-DOS is approximately 10% lower than that of the two traditional algorithms in the 'SRBCT' dataset. In the '9 Tumor' dataset, the error rate of our proposed FS-DOS is lower than that of MDL+LFS by approximately 12% and lower than that of MDL+CON by approximately 25%. Regarding the error rate in the 'Brain Tumor1' dataset, FS-DOS is lower than that of the two traditional methods by approximately 20%. Thus, our proposed FS-DOS shows obvious advantages over these two traditional algorithms in terms of the error rate in most of the adopted datasets.

In summary, FS-DOS wins 21 times, draws 3 times, and loses 6 times out of 30 comparisons compared to MDL+LFS and MDL+CON in the 15 adopted datasets. The simulation results listed in Table 5 demonstrate that our proposed FS-DOS can achieve low error rates by only selecting a few features. Therefore, FS-DOS has significant advantages in handling high-dimensional datasets.

Table 5 Comparison between FS-DOS and traditional methods

Dataset	Algorithm	Size	Error	S	Dataset	Algorithm	Size	Error	S
SRBCT	MDL+LFS	6.1	11.25	-	Prostate6033	MDL+LFS	5.2	**11.33**	+
	MDL+CON	**4.3**	14.17	-		MDL+CON	**4.8**	13.93	≈
	FS-DOS	13.47	**3.88**			FS-DOS	24.23	13.75	
Lymphoma	MDL+LFS	5.6	**1.62**	+	Adenocarcinoma	MDL+LFS	4	40.97	-
	MDL+CON	4.8	3.23	+		MDL+CON	**3.6**	53.32	-
	FS-DOS	**3.2**	7.33			FS-DOS	28.26	**35.38**	
Breast3	MDL+LFS	7.5	43.55	-	Brain Tumor 2	MDL+LFS	5.6	46.67	-
	MDL+CON	**6.7**	66.24	-		MDL+CON	**4.7**	38.33	-
	FS-DOS	113.3	**40.59**			FS-DOS	43.2	**27.50**	
Nci	MDL+LFS	5.8	**27.08**	+	Prostate	MDL+LFS	4.9	26.83	-
	MDL+CON	**5.6**	47.22	-		MDL+CON	**4.7**	29.50	-
	FS-DOS	179.77	29.60			FS-DOS	8.2	**13.67**	
Leukemia 1	MDL+LFS	4.8	18.61	-	Leukemia2	MDL+LFS	4.3	10.00	≈
	MDL+CON	**3**	**10.83**	+		MDL+CON	**3**	14.44	-
	FS-DOS	29.2	16.67			FS-DOS	14.5	**9.44**	
DLBCL	MDL+LFS	4	26.00	-	11 Tumor	MDL+LFS	14.3	38.29	-
	MDL+CON	**3.4**	**7.50**	+		MDL+CON	**9.4**	46.17	-
	FS-DOS	13.7	10.00			FS-DOS	94.1	**21.36**	
9 Tumor	MDL+LFS	12.6	58.33	-	Lung Cancer	MDL+LFS	12.2	**19.45**	≈
	MDL+CON	**7.6**	71.67	-		MDL+CON	**6.6**	29.50	-
	FS-DOS	98.9	**46.50**			FS-DOS	33.8	20.37	
Brain Tumor 1	MDL+LFS	9.9	40.83	-					
	MDL+CON	**6.2**	44.58	-					
	FS-DOS	55.5	**23.57**						

5.4.Effectiveness of the proposed operators in FS-DOS

Research Question 3: Would it be better than FS-DOS to use BDE or QS operators alone for high-dimensional feature selection problems?

To validate the effectiveness of our proposed operator, two FS-DOS variants are designed here for performance comparison, abbreviated as FS-QS and FS-BDE. Note that FS-QS only uses the QS operator, while FS-BDE only adopts the BDE operator. We plot the final population of the training set obtained by running our proposed FS-DOS and its two variants on the 15 adopted datasets in Fig. 6, where the blue pentagram, red inverted triangle, and green dot indicate the solutions obtained by FS-DOS, FS-BDE, and FS-QS, respectively.

It is clear from Fig. 6 that when only one operator is used in the evolutionary process, their obtained results are inferior to those of our proposed FS-DOS combining these two complementary search operators. In particular, it is clear that when only the BDE operator is used, the dimensionality reduction ability of FS-BDE is poor, but the error rate of FS-BDE is not poor. This phenomenon verifies that the proposed BDE operator has a strong global search ability and can maintain the diversity of the population. On the other hand, FS-QS shows some advantages in dimensionality reduction, while its error rates are much higher than those of FS-DOS in most of the datasets used. Therefore, we can learn that the QS operator is effective in dimensionality reduction but is insufficient in terms of reducing the error rate.

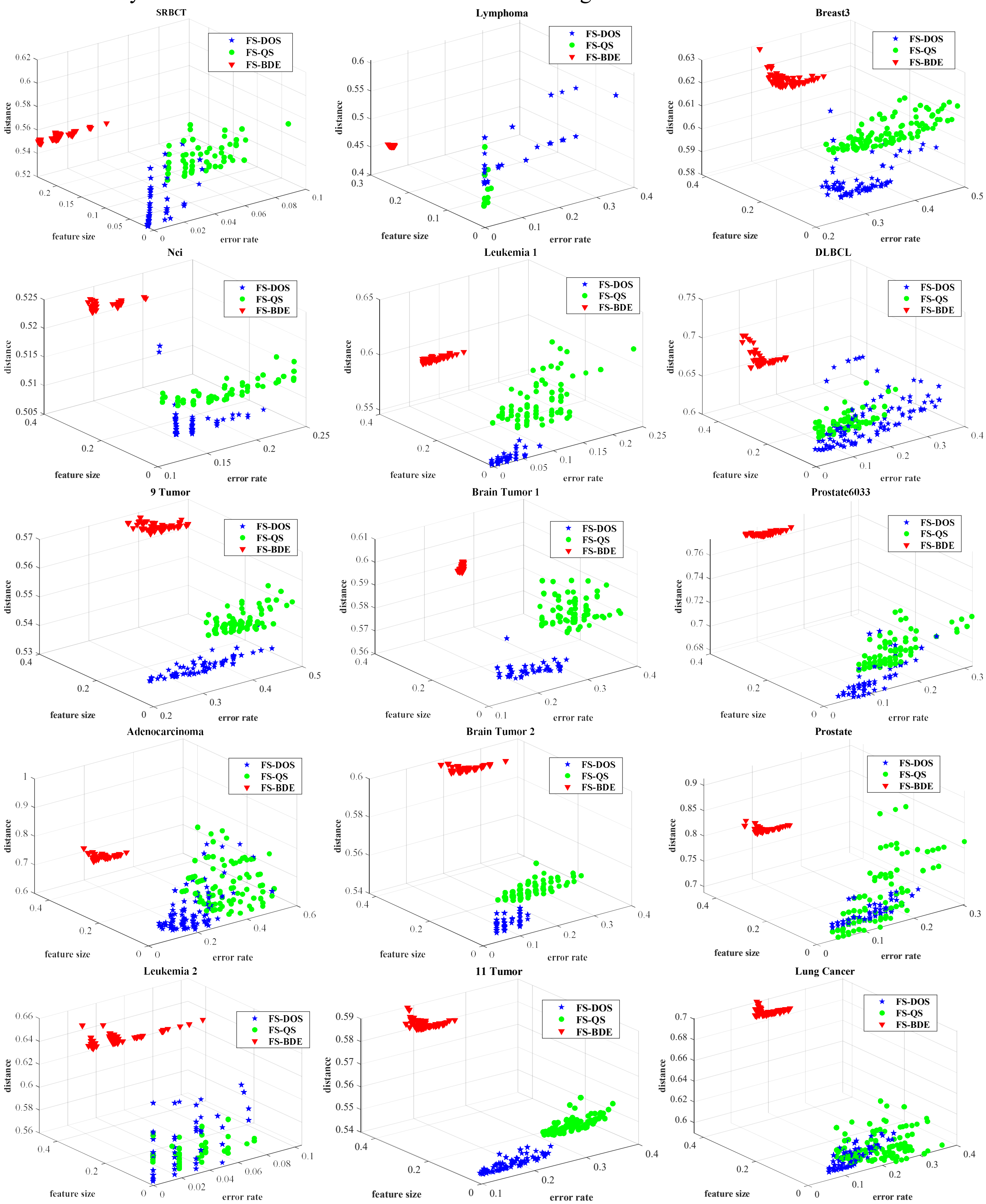


Fig. 6. The results obtained by FS-DOS, FS-QS, and FS-BDE on the adopted 15 datasets at last generation.

To further verify the validity of the proposed operator, the comparison results of FS-DOS, FS-QS, and FS-BDE on the test set are given in Table 6. As seen from the table, in terms of size, FS-QS is effective for dimensionality reduction, and the number of features selected for FS-BDE is almost half of the original dataset. In terms of error rates, FS-BDE has similar error rates to FS-DOS on the ‘SRBCT’, ‘Nci’, ‘Brain

Tumor 1' and 'Lung Cancer' datasets and even better error rates than FS-DOS on the 'Lymphoma' dataset. FS-QS has a similar error rate to FS-DOS on only three datasets ('Prostate6033', 'Brain Tumor 2', 'Leukemia2') and is significantly worse than FS-DOS on the remaining datasets. In summary, the experimental results demonstrate that the QS operator shows advantages for tackling datasets with high-dimensional features, while the BDE operator shows strong exploration ability when the dimensionality of the dataset is reduced to a specific level (i.e., low-dimensional features). Therefore, these two complementary search operators contained in our proposed method can achieve a good tradeoff between exploration and exploitation during the search process.

Table 6 Comparison between FS-DOS, FS-QS and FS-BDE

Dataset	Algorithm	Size	Error	S	Dataset	Algorithm	Size	Error	S
SRBCT	FS-DOS	**13.47**	3.82		Prostate6033	FS-DOS	24.23	**13.75**	
	FS-QS	15.7	9.17	-		FS-QS	**11.4**	14.67	≈
	FS-BDE	572.2	**3.75**	≈		FS-BDE	2325.6	15.67	-
Lymphoma	FS-DOS	3.2	7.33		Adenocarcinoma	FS-DOS	28.26	**35.38**	
	FS-QS	**2.8**	13.5	-		FS-QS	**5.6**	48.51	-
	FS-BDE	1184.8	**0.83**	+		FS-BDE	4043.5	37.02	-
Breast3	FS-DOS	113.3	**40.59**		Brain Tumor 2	FS-DOS	43.2	**27.50**	
	FS-QS	**42.3**	45.11	-		FS-QS	**24.6**	28.23	≈
	FS-BDE	1863.1	42.94	-		FS-BDE	4386.6	37.5	-
Nci	FS-DOS	179.77	**29.60**		Prostate	FS-DOS	8.2	**13.62**	
	FS-QS	**84.7**	34.51	-		FS-QS	**5.2**	18.5	-
	FS-BDE	1875.4	30.51	≈		FS-BDE	4270.8	19	-
Leukemia 1	FS-DOS	29.2	**16.67**		Leukemia2	FS-DOS	14.5	**9.44**	
	FS-QS	**28.5**	18.75	-		FS-QS	**12.3**	10	≈
	FS-BDE	1853.3	19.17	-		FS-BDE	4863.2	11.12	-
DLBCL	FS-DOS	13.7	**10.00**		11 Tumor	FS-DOS	94.1	**21.36**	
	FS-QS	**12.4**	14.50	-		FS-QS	**52**	27.30	-
	FS-BDE	2129.4	15.17	-		FS-BDE	5375.6	24.28	-
9 Tumor	FS-DOS	98.9	**46.50**		Lung Cancer	FS-DOS	33.8	**20.37**	
	FS-QS	**53.1**	57.04	-		FS-QS	**15.4**	27.68	-
	FS-BDE	2114.4	49.95	-		FS-BDE	5392.6	21.26	≈
Brain Tumor 1	FS-DOS	55.5	23.57						
	FS-QS	**21.1**	25.92	-					
	FS-BDE	1833.6	**20.42**	≈					

In summary, these two search operators have their own advantages and disadvantages. No single search operator can achieve a good tradeoff between dimensionality reduction and error rates. To this end, it is reasonable to design a dynamic operator selection strategy to combine these two complementary search operators. As shown in Fig. 6, our proposed FS-DOS using the designed dynamic operator selection (DOS) strategy outperforms FS-BDE and FS-QS, which only use a single search operator, as the final solution set obtained by our proposed FS-DOS shows that FS-DOS can achieve good performance in both dimensionality reduction and error rates.

5.5. The effectiveness of the DOS strategy

Research Question 4: Is there a better value selection for the DOS strategy?

To verify the effectiveness of our proposed DOS strategy, we employ four DOS variants here for performance comparison. Specifically, DOS-0.2, DOS-0.4, DOS-0.6 and DOS-0.8 represent DOS with fixed selection probabilities of 0.2, 0.4, 0.6 and 0.8, respectively. The results of error rates and selected feature sizes obtained by our proposed DOS strategy and its four variants are shown in boxplots in Fig. 7 and Fig. 8, respectively. Due to space limitations, only the comparison results of the 'Breast3', ‘Prostate6033’, ‘Adenocarcinoma’, ‘Prostate’, ‘11 Tumor’ and ‘Lung Cancer’ datasets are presented in this paper.

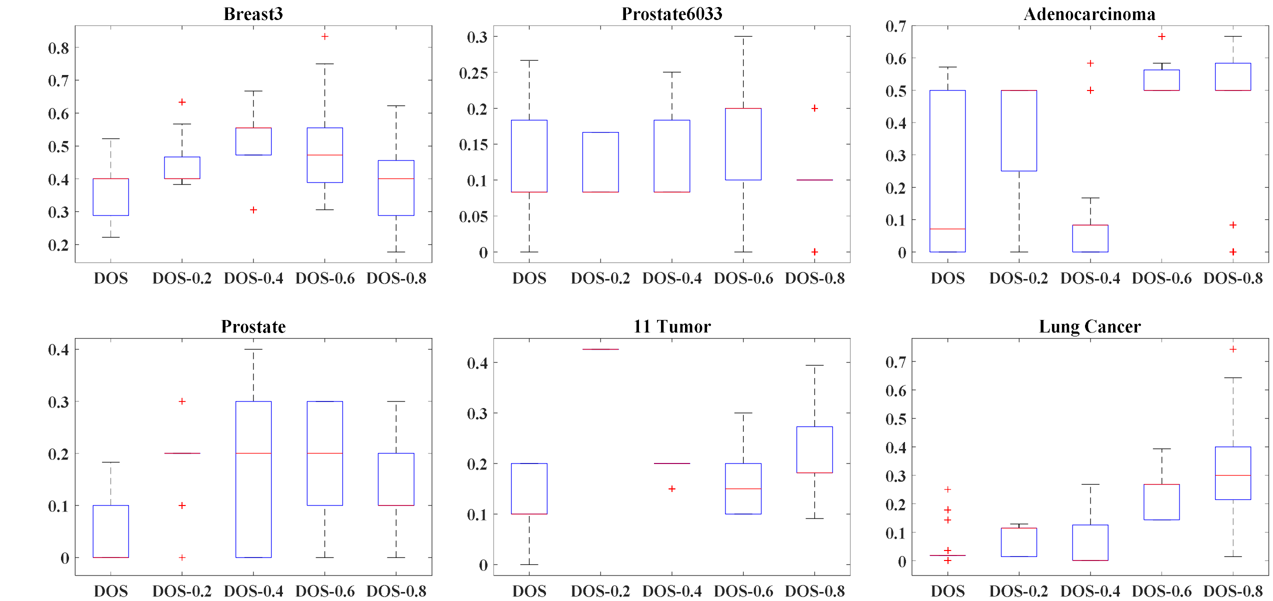


Fig. 7 Box plots of error rates obtained by our DOS and its four variants with fixed selection probabilities.

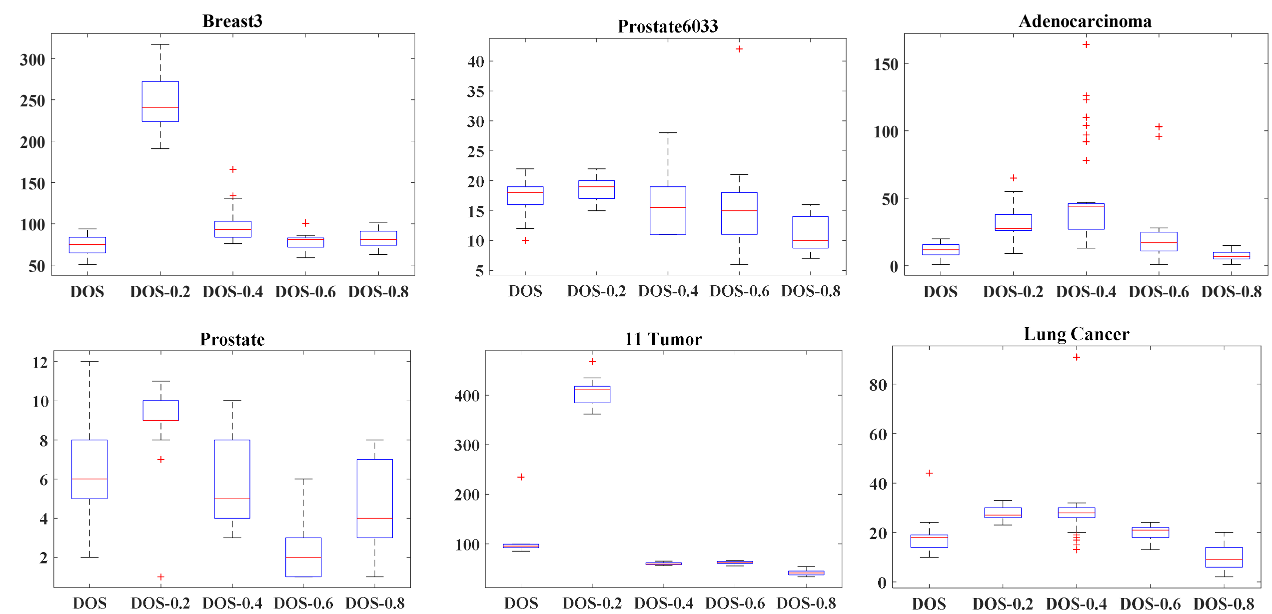


Fig. 8 Box plots of selected feature sizes obtained by our DOS and its four variants with fixed selection probabilities.

As shown in Fig. 7, our designed DOS strategy obtains much lower and more stable error rates when compared to its four variants. Moreover, our proposed DOS strategy also shows obvious superiority in terms of the selected feature sizes. Specifically, compared to DOS-0.2 and DOS-0.4, which have low fixed selection probabilities, the proposed DOS strategy has a clear advantage in terms of the error rate in all datasets except the ‘Lung Cancer’ dataset and shows significant advantages in terms of the selected feature sizes in all datasets except the ‘Prostate6033’ and ‘11 Tumors’ datasets. On the other hand, compared to DOS-0.6 and DOS-0.8, which have high fixed selection probabilities, our designed DOS strategy also has obvious advantages in terms of the error rate in all datasets.

In summary, the simulation results demonstrate that different probabilities assigned for selecting different search operators have various performances in different datasets. Notably, the higher the selection probability, the smaller the selected feature size, and the lower the accuracy, and vice versa. Therefore, the selection probability for the search operator has a crucial impact on the performance of our method on different datasets. Hence, it is necessary to design an effective dynamic operator selection strategy that can better utilize these two complementary search operators, aiming to improve the robustness and scalability of the algorithm when solving various high-dimensional datasets.

6. Conclusions and future work

This paper has proposed an efficient multiobjective optimization algorithm with a dynamic operator selection strategy (called FS-DOS) for feature selection in high-dimensional classification datasets. To achieve fast convergence of the algorithm, we designed a quick search (QS) operator to quickly eliminate redundant features, but this operator might easily become trapped in local optimum. Thus, we also proposed a modified BDE operator with strong global search ability for diversity maintenance. Then, we proposed a resource allocation-guided dynamic operator selection strategy, called DOS, to make better use of the above two complementary operators, which can select the most appropriate search operator for each solution according to its corresponding performance improvement on aggregation objective values. To verify the effectiveness and robustness of our proposed FS-DOS, a large number of experiments were performed on 15 high-dimensional medical-related datasets. Simulation results validated that FS-DOS performs better than several advanced FS algorithms in most cases when considering the error rates and selected feature size. Moreover, ablation experiments further validated the effectiveness of our proposed DOS strategy. Therefore, this paper not only proposes novel technologies but also performs interdisciplinary research with medical datasets.

Although the experimental results have validated the effectiveness of FS-DOS, it still has some limitations that should be further improved in our future work. First, since a wrapper approach requires more computational costs to evaluate solutions, we should design an effective search strategy to reduce the number of evaluations. Second, a coevolutionary algorithm should be considered to combine different search operators in a cooperative manner. Third, we should consider dividing the search space into different subspaces and apply the idea of transfer optimization to search different subspaces to improve our performance in solving the high-dimensional FS problem with huge search spaces.

Acknowledgements

This work was supported by the Shenzhen Science and Technology Program under Grants JCYJ20220531101411027 and JCYJ20190808174801673; in part by the Shenzhen Science and Technology Innovation Commission under Grant 20200805142159001; in part by the Guangdong Regional Joint Foundation Key Project under Grant 2022B1515120076; and in part by Natural Science Foundation of Guangdong Province under Grant 2023A1515011238, Prof. Carlos A. Coello Coello was partially supported by CONACyT grant no. 2016-01-1920 (Investigacion en Fronteras de la Ciencia 2016).

Reference

[1] M. Dash, H. Liu, Feature selection for classification, Intelligent Data Analysis, 1 (1997) 131-156.

[2] F. Di Martino, S. Senatore, Balancing the user-driven feature selection and their incidence in the clustering structure formation, Applied Soft Computing, 98 (2021) 106854.

[3] M. Mokhtia, M. Eftekhari, F. Saberi-Movahed, Feature selection based on regularization of sparsity based regression models by hesitant fuzzy correlation, Applied Soft Computing, 91 (2020) 106255.

[4] X. Na, M. Han, W. Ren, K. Zhong, Modified BBO-Based multivariate time-series prediction system with feature subset selection and model parameter optimization, IEEE Transactions on Cybernetics, 52 (2020) 1-11.

[5] C. Yao, Y.-F. Liu, B. Jiang, Jungong Han, Junwei Han, LLE score: A new filter-based unsupervised feature selection method based on nonlinear manifold embedding and its application to image recognition, IEEE Transactions on Image Processing, 26 (2017) 5257-5269.

[6] Z. Hua, J. Zhou, Y. Hua, W. Zhang, Strong approximate Markov blanket and its application on filter-based feature selection, Applied Soft Computing, 87 (2020) 105957.

[7] S. Wang, W. Zhu, Sparse graph embedding unsupervised feature selection, IEEE Transactions on Systems, Man, and Cybernetics: Systems, 48 (2018) 329-341.

[8] S. Maldonado, J. López, Dealing with high-dimensional class-imbalanced datasets: embedded feature selection for SVM classification, Applied Soft Computing, 67 (2018) 94-105.

[9] B. Tran, B. Xue, M. Zhang, A new representation in PSO for discretization-based feature selection, IEEE Transactions on Cybernetics, 48 (2018) 1733-1746.

[10] B.H. Nguyen, B. Xue, P. Andreae, H. Ishibuchi, M. Zhang, Multiple reference points-based decomposition for multiobjective feature selection in classification: static and dynamic mechanisms, IEEE Transactions on Evolutionary Computation, 24 (2020) 170–184..

[11] A.W. Whitney, A direct method of nonparametric measurement selection, IEEE Transactions on Computers, C-20 (1971) 1100-1103.

[12] T. Marill, D. Green, On the effectiveness of receptors in recognition systems, IEEE Transactions on Information Theory, 9 (1963) 11-17.

[13] S.R. Safavian, D. Landgrebe, A survey of decision tree classifier methodology, IEEE Transactions on Systems, Man, and Cybernetics, 21 (1991) 660-674.

[14] Md.M. Kabir, Md. Shahjahan, K. Murase, 2011. A new local search based hybrid genetic algorithm for feature selection. Neurocomputing, 74 (2011) 2914-2928.

[15] B. Xue, M. Zhang, W.N. Browne, Particle swarm optimization for feature selection in classification: a multi-objective approach, IEEE Transactions on Cybernetics, 43 (2013) 1656-1671.

[16] E. Zorarpacı, S.A. Özel, A hybrid approach of differential evolution and artificial bee colony for feature selection, Expert Systems with Applications, 62 (2016) 91-103.

[17] H.B. Nguyen, B. Xue, I. Liu, P. Andreae, M. Zhang, New mechanism for archive maintenance in PSO-based multi-objective feature selection, Soft Comput, 20 (2016) 3927-3946.

[18] Y. Zhang, D. Gong, J. Cheng, Multi-Objective Particle Swarm Optimization Approach for Cost-Based Feature Selection in Classification, IEEE/ACM Transactions on Computational Biology and Bioinformatics, 14 (2017) 64-75.

[19] Y. Xue, Y. Tang, X. Xu, J. Liang, F. Neri, Multi-Objective Feature Selection With Missing Data in Classification, IEEE Transactions on Emerging Topics in Computational Intelligence, 6 (2021) 1-10.

[20] K. Deb, H. Jain, An evolutionary many-objective optimization algorithm using reference-point-based nondominated sorting approach, Part I: solving problems with box constraints, IEEE Transactions on Evolutionary Computation, 18 (2014) 577-601.

[21] R. Armina, A.M. Zain, N.A. Ali, R. Sallehuddin, A review on missing value estimation using imputation algorithm, J. Phys.: Conf. Ser, 892 (2017) 012004.

[22] C. Bao, D. Gao, W. Gu, L. Xu, E. Goodman, A new adaptive decomsposition-based evolutionary algorithm for multi- and many-objective optimization, Expert Systems with Applications, 213 (2023) 119080.

[23] Z. Wang, M. Gong, P. Li, J. Gu, W. Tian, A hypervolume distribution entropy guided computation resource allocation mechanism for the multiobjective evolutionary algorithm based on decomposition. Applied Soft Computing, 116 (2022) 108297.

[24] L. Li, Q. Lin, Z. Ming, Multi-objective optimization using self-organizing decomposition and its applications to crashworthiness design, Applied Soft Computing, 101 (2021) 107002.

[25] F. Cheng, F. Chu, Y. Xu, L. Zhang, A steering-matrix-based multiobjective evolutionary algorithm for high-dimensional feature selection, IEEE Transactions on Cybernetics, (2021) 1-14. https://doi.org/10.1109/TCYB.2021.3053944.

[26] Y. Zhou, W. Zhang, J. Kang, X. Zhang, X. Wang, A problem-specific non-dominated sorting genetic algorithm for supervised feature selection, Information Sciences, 547 (2021) 841-859.

[27] Y. Zhang, Y. Wang, D. Gong, X. Sun, Clustering-guided particle swarm feature selection algorithm for high-dimensional imbalanced data with missing values, IEEE Transactions on Evolutionary Computation, (2021) 1-1. https://doi.org/10.1109/TEVC.2021.3106975.

[28] M. Gutlein, E. Frank, M. Hall, A. Karwath, Large-scale attribute selection using wrappers, in: 2009 IEEE Symposium on Computational Intelligence and Data Mining. Presented at the 2009 IEEE Symposium on Computational Intelligence and Data Mining, (2009) 332-339. https://doi.org/10.1109/CIDM.2009.4938668.

[29] L.-Y. Chuang, H.-W. Chang, C.-J. Tu, C.-H. Yang, Improved binary PSO for feature selection using gene expression data, Computational Biology and Chemistry, 32 (2008) 29-38.

[30] D.A. Gastelum Chavira, J.C. Leyva Lopez, J.J. Solano Noriega, O. Ahumada Valenzuela, P.A. Alvarez Carrillo, A credit ranking model for a parafinancial company based on the ELECTRE-III method and a multiobjective evolutionary algorithm, Applied Soft Computing, 60 (2017) 190-201.

[31] Z.K. Wang, H.L. Zhen, J.D. Deng, Q.F. Zhang, X.J. Li, M.X. Yuan, J. Zeng, Multiobjective optimization-aided decision-making system for large-scale manufacturing planning, IEEE Transactions on Cybernetics, 52 (2022) 8326-8339.

[32] F. Cheng, J. Chen, J. Qiu, L. Zhang, A subregion division based multi-objective evolutionary algorithm for SVM training set selection, Neurocomputing, 394 (2020) 70-83.

[33] X. Zhang, F. Duan, L. Zhang, F. Cheng, Y. Jin, K. Tang, Pattern recommendation in task-oriented applications: a multi-objective perspective [Application Notes], IEEE Computational Intelligence Magazine, 12 (2017) 43-53.

[34] X. Zhang, K. Zhou, H. Pan, L. Zhang, X. Zeng, Y. Jin, A network reduction-based multiobjective evolutionary algorithm for community detection in large-scale complex networks, IEEE Transactions on Cybernetics, 50 (2020) 703-716.

[35] Z.K. Wang, Y.S. Ong, J.Y. Sun, A. Gupta, Q.F. Zhang, A generator for multiobjective test problems with difficult-to-approximate Pareto front boundaries, IEEE Transactions on Evolutionary Computation, 23 (2019) 556-571.

[36] L. Li, Q. Lin, Z. Ming, K. Wong, M. Gong, and C. Coello Coello, An immune-inspired resources allocation strategy for many-objective optimization, IEEE Transactions on Systems, Man, and Cybernetics: Systems, doi: 10.1109/TSMC.2022.3221466.

[37] Q.F. Zhang, H. Li, MOEA/D: A multiobjective evolutionary algorithm based on decomposition, IEEE Transactions on Evolutionary Computation, 11 (2007) 712-731.

[38] M. Li, X. Yao, What weights work for you? adapting weights for any pareto front shape in decomposition-based evolutionary multiobjective optimisation, Evolutionary Computation, 28 (2020) 227-253.

[39] R. Tanabe, H. Ishibuchi, A framework to handle multimodal multiobjective optimization in decomposition-based evolutionary algorithms, IEEE Transactions on Evolutionary Computation, 24 (2020) 720-734.

[40] X. Cai, Z. Yang, Z. Fan, Q. Zhang, Decomposition-based-sorting and angle-based-selection for evolutionary multiobjective and many-objective optimization, IEEE Transactions on Cybernetics, 47 (2017) 2824-2837.

[41] Q. Song, J. Ni, G. Wang, A fast clustering-based feature subset selection algorithm for high-dimensional data, IEEE Transactions on Knowledge and Data Engineering, 25 (2013) 1-14.

[42] X.-F. Song, Y. Zhang, D.-W. Gong, X.-Z. Gao, A fast hybrid feature selection based on correlation-guided clustering and particle swarm optimization for high-dimensional data, IEEE Transactions on Cybernetics, (2021) 1-14. https://doi.org/10.1109/TCYB.2021.3061152.

[43] X. Song, Y. Zhang, D. Gong, X. Sun, Feature selection using bare-bones particle swarm optimization with mutual information, Pattern Recognition, 112 (2021) 107804.

[44] O. Tarkhaneh, T.T. Nguyen, S. Mazaheri, A novel wrapper-based feature subset selection method using modified binary differential evolution algorithm, Information Sciences, 565 (2021) 278–305.

[45] Y. Zhang, D. Gong, X. Gao, T. Tian, X. Sun, Binary differential evolution with self-learning for multi-objective feature selection. Information Sciences, 507 (2020) 67-85.

[46] B. Tran, B. Xue, M. Zhang, Variable-length particle swarm optimization for feature selection on high-dimensional classification, IEEE Transactions on Evolutionary Computation, 23 (2019) 473-487.

[47] Y. Tian, C. Lu, X. Zhang, K. C. Tan and Y. Jin, Solving Large-Scale Multiobjective Optimization Problems With Sparse Optimal Solutions via Unsupervised Neural Networks, IEEE Transactions on Cybernetics,51 (2021) 3115-3128.

[48] Xue, Y., Zhu, H., Neri, F. A feature selection approach based on NSGA-II with ReliefF. Applied Soft Computing, 134 (2023), 109987.

[49] Pan, H., Chen, S., Xiong, H. A high-dimensional feature selection method based on modified Gray Wolf Optimization. Applied Soft Computing, 135 (2023), 110031.

[50] L. Li., M. Xuan., Q. Lin., M. Jiang., Z. Ming., K. C. Tan. An Evolutionary Multitasking Algorithm with Multiple Filtering for High-Dimensional Feature Selection. IEEE Transactions on Evolutionary Computation, (2023). doi: 10.1109/TEVC.2023.3254155.

[51] G. Patterson, M. Zhang, Fitness functions in genetic programming for classification with unbalanced data, in: Orgun, M.A., Thornton, J. (Eds.), AI 2007: Advances in Artificial Intelligence, Lecture Notes in Computer Science, Springer, Berlin, Heidelberg, (2007) pp. 769-775.

[52] U. Fayyad, K. Irani, Multi-interval discretization of continuous-valued attributes for classification learning, in Proc. 13th Int. Joint Conf. Artif. Intell., vol. 2. Chambéry, France, (1993) 1022-1027..

[53] M. Hall et al, The WEKA Data Mining Software: An Update, in ACM SIGKDD Explorations Newslett, 11(2009) 10-18, 2009.

[54] N. Cliff, Ordinal Methods for Behavioral Data Analysis, Psychology Press, New York, 2014.