



ARTICLE

Leveraging Opposition-Based Learning in Particle Swarm Optimization for Effective Feature Selection

Fei Yu^{1,2,3,*}, Zhenya Diao^{1,2}, Hongrun Wu^{1,2,*}, Yingpin Chen^{1,3}, Xuewen Xia^{1,2} and Yuanxiang Li^{2,3,4}

¹School of Physics and Information Engineering, Minnan Normal University, Zhangzhou, 363000, China

²Key Lab of Intelligent Optimization and Information Processing, Minnan Normal University, Zhangzhou, 363000, China

³Key Laboratory of Light Field Manipulation and System Integration Applications in Fujian Province, Minnan Normal University, Zhangzhou, 363000, China

⁴School of Computer Science, Wuhan University, Wuhan, 430072, China

*Corresponding Authors: Fei Yu. Email: yufei@whu.edu.cn; Hongrun Wu. Email: wuhongrun@whu.edu.cn

Received: 30 August 2025; Accepted: 18 November 2025; Published: 10 February 2026

ABSTRACT: Feature selection serves as a critical preprocessing step in machine learning, focusing on identifying and preserving the most relevant features to improve the efficiency and performance of classification algorithms. Particle Swarm Optimization has demonstrated significant potential in addressing feature selection challenges. However, there are inherent limitations in Particle Swarm Optimization, such as the delicate balance between exploration and exploitation, susceptibility to local optima, and suboptimal convergence rates, hinder its performance. To tackle these issues, this study introduces a novel Leveraged Opposition-Based Learning method within Fitness Landscape Particle Swarm Optimization, tailored for wrapper-based feature selection. The proposed approach integrates: (1) a fitness-landscape adaptive strategy to dynamically balance exploration and exploitation, (2) the lever principle within Opposition-Based Learning to improve search efficiency, and (3) a Local Selection and Re-optimization mechanism combined with random perturbation to expedite convergence and enhance the quality of the optimal feature subset. The effectiveness of is rigorously evaluated on 24 benchmark datasets and compared against 13 advanced metaheuristic algorithms. Experimental results demonstrate that the proposed method outperforms the compared algorithms in classification accuracy on over half of the datasets, whilst also significantly reducing the number of selected features. These findings demonstrate its effectiveness and robustness in feature selection tasks.

KEYWORDS: Feature selection; fitness landscape; opposition-based learning; principle of the lever; particle swarm optimization

1 Introduction

Feature selection (FS), a vital data preprocessing technique, aims to identify and select the most informative subset of features from a redundant dataset. This process helps to reduce model complexity, prevent overfitting, enhance model generalization, and mitigate computational costs [1]. FS is a critical step in both machine learning [2] and data mining [3], serving as a key method for handling redundant data. In a dataset with n dimensions, the possible combinations of features reach 2^n . This exponential growth illustrates how the number of possible feature combinations increases dramatically as the dimensionality of the dataset rises. Consequently, feature selection faces the severe challenge of the “curse of dimensionality” [4].

To address the “curse of dimensionality” in FS, researchers have proposed numerous effective methods. These methods are typically categorized into three groups based on distinct evaluation criteria: filter-based



approaches, wrapper-based approaches, and embedded approaches [5]. Filter-based approaches [6,7] depend exclusively on the statistical properties of features (e.g., variance, correlation, consistency, information gain, chi-square test), by scoring each feature and then selecting those with the highest scores. This method generally uses fewer computational resources but typically yields lower classification accuracy compared to wrapper-based approaches. Wrapper-based approaches [8,9] integrate the feature selection process into the model training (e.g., *K*-nearest neighbor (KNN), support vector machines (SVMs)), selecting features based on the model's performance. This method typically requires more computational resources but achieves better classification accuracy. Embedded approaches [10,11] perform feature selection simultaneously during model training (e.g., Lasso regression, decision trees (DT), random forests (RF)). Generally, wrapper-based and embedded methods achieve higher classification accuracy than filter methods, nevertheless, this frequently incurs higher computational costs and diminishes generalization capability.

Conventional feature selection methods, despite their efficacy in numerous applications, continue to encounter challenges such as elevated computational costs and the risk of neglecting feature combinations that substantially enhance model performance. Additionally, using wrapper-based and embedded approaches to directly optimize model performance often leads to overfitting and weak generalization ability. In contrast, evolutionary computation (EC)-based feature selection possesses global search capabilities, which help avoid local optima and identify better feature combinations, effectively handling high-dimensional and complex data [12]. Furthermore, EC-based wrapper algorithms can be classified by the EC paradigms they use, such as genetic algorithms (GA) [13], particle swarm optimization (PSO) [14,15], and genetic programming (GP) [16,17].

In recent years, several prominent researchers have made significant contributions to exploring feature subsets using EC-based feature selection methods. Mirjalili et al. introduced a heuristic algorithm for engineering design problems, referred to as the Salp Swarm Algorithm (SSA) [18]. Subsequently, Tubishat et al. enhanced the Salp Swarm Algorithm, using the opposition-based learning (OBL) [19] strategy, resulting in the Improved Salp Swarm Algorithm (ISSA) [20]. Shambour et al. developed a new Flower Pollination Algorithm (FPA) variant that improves convergence speed and solution quality, known as modified global FPA (mgFPA) [21]. Similar swarm intelligence algorithms have also been proposed, such as the Whale Optimization Algorithm (WOA) by Mirjalili and Lewis [22]. Ouadfel and Abd Elaziz [23] employed an enhanced crow search algorithm (ECSA) for feature subset selection. Another category of heuristic algorithms has been applied to the FS problem. For example, Too and Rahim Abdullah proposed binary variants of Atom Search Optimization (BASO) [1] for wrapper feature selection. These evolutionary algorithms have demonstrated strong competitiveness in obtaining effective feature subsets. However, many algorithms suffer from inadequate search capabilities when finding feature subsets. PSO, owing to its simplicity in implementation, fast convergence, minimal parameter adjustment, and strong global search ability, has emerged as a favored approach in FS [24]. Initially, a novel binary PSO was proposed by Khanesar et al. [25] for solving function optimization problems. Shortly thereafter, Chuang et al. [26] utilized binary PSO for feature selection, specifically to manage gene expression data. The emergence of diverse swarm intelligence algorithms is primarily defined by a population information-sharing mechanism akin to that of PSO. This resemblance is attributable to the fact that PSO is an algorithm derived from natural predatory behaviors. Evidently, the majority of swarm intelligence algorithms developed thus far incorporate information exchange rooted in predatory behaviors [27], which can be regarded as adaptations of the PSO algorithm tailored to the specific behaviors of different biological species. Since PSO faces issues like sensitivity to parameter tuning, and performance degradation from the “curse of dimensionality”, finding the optimal feature subset becomes increasingly challenging. This paper presents an innovative PSO algorithm, with the primary contributions outlined as follows.

- By constructing and utilizing fitness landscapes, the proposed PSO algorithm effectively utilizes the state of the population to adaptively adjust its parameters, which is more in line with the evolutionary behavior of population intelligence and reduces the possibility of missing the optimal subset.
- To mitigate the risk of losing the optimal feature subset due to the curse of dimensionality, a leverage-based opposition learning strategy is proposed. This strategy facilitates the selection of features that may have been previously discarded but are crucial for important feature combinations.
- A novel local selection and re-optimization strategy (LSR) based on Cauchy perturbation is proposed to accelerate the convergence speed of the population towards the optimal feature subset.

The remainder of this paper is organized as follows. [Section 2](#) presents an overview of the fundamental concepts of PSO and FS, along with practical and effective algorithmic strategies. [Section 3](#) introduces a comprehensive overview of the proposed lever-FLPSO. [Section 4](#) showcases the experimental results and their analysis. Lastly, [Section 5](#) concludes the study and outlines potential directions for future research.

2 Background

2.1 The Standard Particle Swarm Optimization

The Particle Swarm Optimization (PSO) is an optimization algorithm based on swarm intelligence, proposed by [28] in 1995. PSO simulates the foraging behavior of birds and employs information sharing among individuals to identify the optimal solution. In PSO, each solution is represented as a particle, which moves through the D -dimensional search space to find the optimal position. The fundamental concept of PSO is that each particle's movement is guided by its personal best position ($\mathbf{PB}_i = [pb_{i,1}, pb_{i,2}, \dots, pb_{i,D}]$, $1 \leq i \leq NP$) and the best position found by the entire swarm ($\mathbf{GB} = [gb_1, gb_2, \dots, gb_D]$). NP represents the total number of particles within the population. The velocity and position update equations for each particle are expressed as follows:

$$v_{i,j}^{t+1} = w \cdot v_{i,j}^t + c_1 \cdot r_{1,j} \cdot (pb_{i,j} - x_{i,j}^t) + c_2 \cdot r_{2,j} \cdot (gb_j - x_{i,j}^t) \quad (1)$$

$$x_{i,j}^{t+1} = x_{i,j}^t + v_{i,j}^{t+1} \quad (2)$$

where, the velocity update formula [Eq. \(1\)](#) guides the velocity update for each particle i in dimension j , where $v_{i,j}^{t+1}$ denotes the velocity of particle i in dimension j at time $t + 1$. w is the inertia weight, c_1 and c_2 are individual and social learning factors, $r_{1,j}$ and $r_{2,j}$ are random variables uniformly distributed in the range $[0, 1]$, $pb_{i,j}$ represents the personal best position of particle i in dimension j , and gb_j is the value of the global best position in dimension j . The position update formula [Eq. \(2\)](#) details the procedure for updating the position of particle i in dimension j , where $x_{i,j}^{t+1}$ represents the position of particle i in dimension j at time $t + 1$.

2.2 Fitness Landscape

The fitness landscape (FL) is a conceptual model used in optimization problems to describe the solution space. It is constructed based on the objective function values corresponding to each solution obtained during the evolutionary process. The FL exhibits various topographical features such as ridges, valleys, and basins, which intuitively represents the complexity of the problem and the distribution of the solution space. A complex fitness landscape usually features multiple local optima as well as a global optimum, requiring optimization algorithms to search within this landscape, avoiding local optima and striving to find the global optimum. The fitness landscape plays a crucial role in the design and analysis of optimization algorithms, facilitating the comprehension of the search behavior and performance of the algorithms [29]. In previous studies, various methods have been proposed to utilize the FL to solve optimization problems. Examples

include measures of dispersion metric [30], fitness clouds and length scales [31], and fitness-distance correlation (FDC) [32]. This paper utilizes the FDC to measure the population status.

2.3 Opposition-Based Learning

The OBL [33] is a strategy to enhance the efficiency of optimization algorithms by simultaneously considering the current solution and its opposite, thereby providing a more comprehensive exploration of the search space and increasing the likelihood of finding the global optimum. In OBL, given a current solution x , its opposite solution $\tilde{x}$ can be calculated using the following formula:

$$\tilde{x} = x^L + x^U - x \quad (3)$$

where, x^L and x^U represent the lower and upper bounds of the solution space, respectively. By evaluating both the current solution and its opposite, OBL can more effectively guide the search process, enhancing the convergence speed and optimization performance of the algorithm.

This concept can be extended to a multi-dimensional search space. The OBL is applied to each dimension of the search space, as represented by the following equation:

$$\tilde{x}_j = x_j^L + x_j^U - x_j, \quad j = 1, 2, \dots, D \quad (4)$$

where, x^L and x^U denote the lower and upper bounds of the solution space in dimension j , correspondingly.

2.4 Feature Selection Algorithms Based on Evolutionary Computation Methods

The FS is a critical research topic in machine learning and data mining. When formalized as an optimization problem, feature selection involves selecting the most valuable subset of features $S_d \subseteq \{x_1, x_2, \dots, x_D\}$ ($d < D$) to optimize a given performance metric $H(\cdot)$. The mathematical expression for feature selection is as follows:

$$\begin{aligned} \max \quad & H(S_d) \\ \text{s.t.} \quad & S_d \subseteq \{x_1, x_2, \dots, x_D\} \quad x_j \in \{0, 1\}, \quad j \in \{1, 2, \dots, D\} \end{aligned} \quad (5)$$

where, D is the dimension of the feature, and $H(S)$ is the performance metric of the model constructed using the feature subset S .

Over the past decade, EC algorithms have achieved widespread application in feature selection owing to their strong global search capabilities and adaptability to complex problems. In feature selection tasks, the challenges of high dimensionality, non-linearity, and intricate search spaces often lead to the “curse of dimensionality”, which causes traditional methods to become easily trapped in local optima. EC-based feature selection algorithms, by simulating natural selection and genetic processes, are adept at exploring the entire search space and identifying feature subsets that approach the global optimum. For instance, Ma and Xia [34] developed a tribe competition-based genetic algorithm (TCbGA) for feature selection, where each tribe focuses on a particular region of the solution space. Li et al. [35] proposed a binary differential evolution based on individual entropy for feature selection, utilizing individual entropy to calculate population diversity and designing an entropy-based objective function to evaluate feature subsets. Zhang et al. [16] applied GP to feature selection in dynamic flexible job-shop scheduling. Bi et al. [36] proposed using GP with a small amount of training samples for face image classification. In addition to the aforementioned works, PSO has been utilized for feature selection because of its strong global search capabilities. The application of the PSO in FS primarily involves addressing the enhancement of parameter selection mechanisms and the identification of effective features.

In terms of improving the parameters of the PSO. For instance, Fang et al. introduced an adaptive switching randomly perturbed PSO (ASRPPSO) [37], which leverages its excellent convergence properties for FS and has been applied to industrial data clustering. Azar et al. [38] proposed a parallelized PSO with linearly time-varying acceleration coefficients and inertia weight (PLTVACIW-PSO). Chen et al. [39] introduced a spiral-shaped mechanism with nonlinear inertia weights to optimize PSO (HPSO-SSM), and applied it in FS to enhance the diversity of the search process. Moreover, Gubta and Gubta [40] applied Gaussian fuzzification to improve the acceleration coefficients in PSO (FPSO) for feature selection. Similarly, Beheshti [41] utilized a fuzzy transfer function to control PSO parameter selection (FBPSO), addressing high-dimensional feature selection problems.

On the other hand, improving strategies for selecting effective features. Gao et al. [42] employed a feature grouping strategy grounded in information gain ratio, enabling PSO to perform feature selection effectively. Moradi and Gholampour proposed a hybrid PSO with Local Search strategy (HPSO-LS) [43], which selects less correlated and more salient feature subsets by embedding a local search strategy within PSO. Qiu and Liu [44] invented a novel three-layer PSO (TLPSO) for feature selection, dividing the particles into three layers based on their evolutionary states and treating particles in different layers differently to fully exploit the potential of each particle. In the improvements to PSO, in addition to using search strategies, Too et al. [45] utilized a conditional opposition-based PSO (COPSO) to address the imbalance between exploration and exploitation. Additionally, Tijjani et al. [46] proposed two versions of enhanced binary PSO (EBPSO) for feature selection, known as EBPSO1 and EBPSO2, by applying two novel position update mechanisms to effectively select the optimal feature subset. To tackle the complexity of high-dimensional datasets, Ahadzadeh et al. [47] formulated a simple, fast, and efficient hybrid algorithm (SFE-PSO) that combines two operators, nonselection and selection. Wang et al. designed PSO-based multi-task evolutionary learning (MEL-PSO) [48] to deal with the shortcomings of PSO in effectively utilizing and sharing information within high-dimensional data. Despite numerous improvements to PSO applied to feature selection problems, there remains a significant gap in research on parameter optimization. Additionally, the presence of symmetry within feature sets gives rise to inherent limitations in the basic OBL strategy, thereby reducing its effectiveness. Therefore, this study will build upon this foundation for further in-depth investigation. Finally, Table 1 summarises the contributions and limitations of existing feature selection methods.

Table 1: Summary of PSO and swarm intelligence algorithms in FS problem mentioned in this work

Method	Algorithm	Advantages	Limitations
PSO	TLPSO [44]	Classify the particle swarm into three tiers based on evolutionary states	Particle counts per layer are not adaptive.
	COPSO [45]	Initialisation using OBL mechanism	The evolutionary process lacks an OBL mechanism
	ASRPPSO [37]	Utilising Euclidean distance to partition population states	Evolutionary state parameters lack adaptive properties
	PLTVACIW-PSO [38]	Linearly varying inertia weights and acceleration coefficients	Fixed step size for each parameter variation
	HPSO-SSM [39]	A spiral-shaped mechanism with nonlinear inertia weights to optimize PSO	Location updates involve hyperparameter tuning

(Continued)

Table 1 (continued)

Method	Algorithm	Advantages	Limitations
Swarm	HPSO-LS [43]	Selecting more salient feature subsets by embedding local search strategies	Insensitive to exploration of global features
	SFE-PSO [47]	Combining non-selective and selective operators for rapid feature selection	Non-selective operators may remove highly correlated features
	EBPSO [46]	Determining particle position switching probabilities based on position	The performance depends on the initialisation effectiveness
	MEL-PSO [48]	Utilising multitasking for comprehensive information sharing	Insensitive to class imbalance in data
	BASO [1]	By employing S-shaped and V-shaped functions to binarise the ASO	Algorithm depends critically on the transfer function
Swarm	ECOA [23]	Balancing the exploration and exploitation in adaptive perception	The effect of random initial populations on convergence
	SSA [18]	Efficient initial solution and fast convergence to optimal solution	Insufficient capacity for exploration
	ISSA [20]	The OBL has been introduced in SSA to enhance exploratory skills	Centroid-based OBL lead to a lack of ability to streamline features
	mgFPA [21]	Guiding the exploration process through existing solutions	The discovery process is overly reliant on self-explored solutions

3 Methods

3.1 Construction of Fitness Landscape

This section will focus on defining the states of the population during evolution and explaining how to adaptively set parameters in a given state to enhance the search for optimal feature subsets.

3.1.1 Population Status

In this paper, the fitness-distance correlation (FDC) is employed to characterize the state of the population by calculating the average distance of each particle to all other particles. In Fig. 1, six particles, denoted as $P = \{p_1, p_2, p_3, p_4, p_5, g\}$, are used to represent various aspects of the landscape. Among them, p_1 to p_5 represent the ordinary individuals, while g represents the optimal individual. The lines connecting them symbolize the distances between the particles. As they explore, they will naturally occupy different positions, resulting in distinct fitness values and various population states.

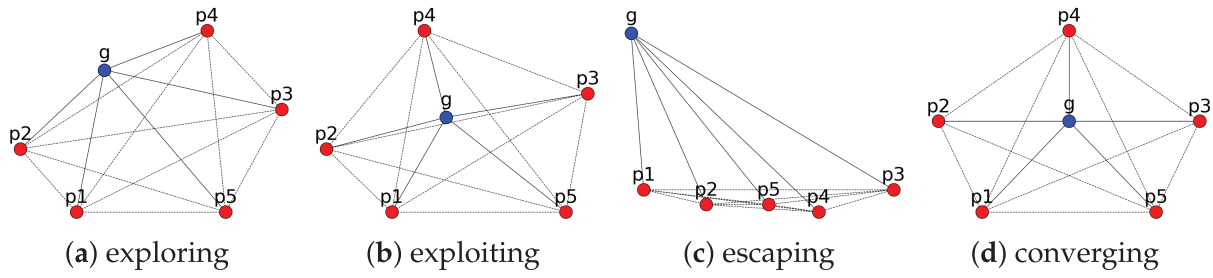


Figure 1: Population status

Throughout the optimization process, it becomes apparent that the particles in a population undergo an initial random state, exploration, exploitation and convergence, as well as escaping state when trapped in local optima, as depicted in Fig. 1. From Fig. 1a, in the initial stage, the population is randomly initialized across the entire search space, exhibiting an exploration state without a clear direction. Nonetheless, as the particles progress in their search, certain individuals start to identify optimal regions, as illustrated in Fig. 1b. Consequently, the collective population of particles gradually converges towards the proximity of the best-performing particle. In Fig. 1c, it is evident that certain particles become ensnared in local optimal states, necessitating intervention to extricate themselves from these local optima, referred to as the escape state. This simple illustration highlights the diverse distribution of the population information throughout the optimization process, and the fitness landscape can clearly depict these states. In the following, a method for characterizing the four states of the population using a fitness landscape is proposed.

The population's position vectors, represented as $X = [X_1, X_2, \dots, X_{NP}]$, the average distance of each individual X_i to all other particles can be calculated using the Euclidean distance, denoted as (d_i) . The formula for calculating d_i is given by Eq. (6).

$$d_i = \frac{1}{NP-1} \sum_{j=1, j \neq i}^{NP} \sqrt{\sum_{k=1}^D (X_{i,k} - X_{j,k})^2} \quad (6)$$

where, NP and D are the population size and dimension, respectively.

Once the average distances of each particle to other particles have been calculated, the average distance of the best particle is denoted as d_g . By comparing all d_i values, we can identify the maximum value ($d_{\max}$) and the minimum value ($d_{\min}$). To represent the four states, introduce a landscape status value λ , which is defined as the Eq. (7).

$$\lambda = \frac{d_g - d_{\min}}{d_{\max} - d_{\min}} \quad (7)$$

where, the result of λ falls within the range $[0, 1]$.

$$\tau_s(\lambda) = \begin{cases} S_1, & \text{if } 0 \leq \lambda \leq 0.3 \\ S_2, & \text{if } 0.3 < \lambda \leq 0.6 \\ S_3, & \text{if } 0.6 < \lambda \leq 0.8 \\ S_4, & \text{if } 0.8 < \lambda \leq 1.0. \end{cases} \quad (8)$$

The identification of the four states is determined by the formula provided in Eq. (8). The function $\tau_s(\lambda)$ into one of the four sets classified into one of the four sets: S_1, S_2, S_3 , and S_4 , representing the

states of convergence, exploitation, exploration, and escaping state, respectively. In our previous work, the different states of the population were determined based on the evolutionary state parameters in the FLPSO [49] algorithm, and it has been demonstrated that the setting of the parameters is effective in population construction.

3.1.2 Self-Adaptive Weighting Strategy

For the PSO, the inertia weight parameter w and acceleration coefficients c_1 and c_2 are crucial control parameters, as the selection of these parameters directly influences the performance and convergence characteristics of the algorithm. The Eq. (9) is employed for the nonlinear variation of the inertia weight parameter w [50], which has proven to be an effective parameter setting in experiments. In addition, for the acceleration coefficients c_1 and c_2 , different strategies are employed with different parameters depending on the state.

$$\omega(\lambda) = \frac{1}{1 + 1.5e^{-2.6\lambda}}, \quad \forall \lambda \in [0, 1]. \quad (9)$$

In the convergence state (S_1), it becomes apparent that the prevalence of local optima hinders the discovery of global optima due to the proximity of particles. To tackle this challenge effectively, it is essential to introduce a certain level of disturbance. This strategic intervention serves to bolster the diversity within the particle population and aids in breaking free from local optima, thus boosting the effectiveness of the optimization process. In common perturbation operations, a novel approach involves utilizing adjustable-intensity Gaussian white noise as described in Eq. (10). This Gaussian perturbation method introduces noise into the linear acceleration parameters (see Eqs. (11) and (12)), thereby enabling a more thorough exploration of the problem space by randomly perturbing the acceleration coefficients. Additionally, an elite learning strategy is implemented during S_1 , where any dimension is chosen from all historically optimal particles to guide the movement of the particle. This further enhances population diversity and the effectiveness of escaping local optima traps.

$$N(x; \mu, \sigma) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2} \quad (10)$$

$$C_p = C_p^{min} + (C_p^{max} - C_p^{min}) \times \frac{(MaxIter - iter)}{MaxIter} \quad (11)$$

$$C_g = C_g^{max} - (C_g^{max} - C_g^{min}) \times \frac{(MaxIter - iter)}{MaxIter} \quad (12)$$

$$\delta(iter) = N(x = iter; \mu, \sigma) \quad (13)$$

where, in Eq. (10) is the probability density function (PDF), μ and σ represent the mean and variance, respectively, where $\mu = 0$, $\sigma = 0.07$, C_p and C_g are linear parameters, where $C_p^{max} = C_g^{max} = 2.5$ and $C_p^{min} = C_g^{min} = 0.5$, $MaxIter$ and $iter$ are the maximum number of iteration cycles and how many times the current cycle, $\delta(\cdot)$ is the disturbance coefficient.

During the exploitation state (S_2), both the global and individual aspects need attention. An adaptive weighting strategy is employed that considers both the distance from the particle to the global best position and the distance from the particle to its personal best position. The adaptive adjustment of acceleration coefficients is described in Eq. (14) using a weighted strategy based on a nonlinear function. This approach enhances convergence speed, facilitating the development of the global optimal solution.

$$F(Z) = \frac{b}{1 + e^{-a \times (Z-c)}} + m \quad (14)$$

$$g_{pi}(iter) = pb_{i,j} - x_{i,j}^t \quad (15)$$

$$g_{gi}(iter) = gb_j - x_{i,j}^t \quad (16)$$

where, in the Eq. (14), $a = 0.000035 \cdot d$, where d represents the size of the search space of the optimization problem. The parameters b , c , and m are set to 0.5, 0, and 1.5, respectively. The vectors $g_{pi}(iter)$ and $g_{gi}(iter)$ represent the difference between the personal best position and the global best position, with detailed usage described in Eqs. (17) and (18).

In the exploration state (S_3), the emphasis is on searching for excellent solutions. A larger cognitive coefficient (c_1) prioritizes particles to move towards their individual best solutions, enabling them to explore solutions within their surrounding environments. On the other hand, a smaller social coefficient (c_2) encourages particles to rely on individual experiences instead of the global experiences of the population. In this configuration, particles are enabled to leverage their accumulated knowledge and engage in local interactions with adjacent particles, thereby facilitating an efficient search for superior solution spaces.

In the escaping state (S_4), it is necessary to use larger values of the social coefficient (c_2) and smaller values of the individual cognitive coefficient (c_1). This ensures that particles are more influenced by the collective experience, guiding them towards the direction of the global optimum. Building on the aforementioned discussion, the parameter settings of RPSO [51] and AWPSO [52] exhibit exceptional performance in the S_1 and S_2 stages. It is clear that the acceleration coefficients of the lever-FLPSO algorithm in each state are presented in Eqs. (17) and (18).

$$c_1 = \begin{cases} C_p + \delta(iter), & S_1 \\ F(g_{pi}(iter)), & S_2 \\ c_1 + \text{rand}(0.05, 0.1), & S_3 \\ c_1 - \text{rand}(0.05, 0.1), & S_4. \end{cases} \quad (17)$$

$$c_2 = \begin{cases} C_g + \delta(iter), & S_1 \\ F(g_{gi}(iter)), & S_2 \\ c_2 - \text{rand}(0.05, 0.1), & S_3 \\ c_2 + \text{rand}(0.05, 0.1), & S_4. \end{cases} \quad (18)$$

3.2 A Novel Opposition-Based Learning Method Based on the Leverage Principle

The lever principle is a fundamental concept in classical mechanics that describes how force can be amplified using a simple mechanical device known as a lever. Archimedes was the first to study and describe the lever principle, which consists of a rigid bar and a fixed fulcrum. When a force exerted at one end of the lever is transmitted through the fulcrum to the other end, it can achieve force amplification or a change in direction, and the following diagram illustrates the lever principle in Fig. 2.

The mathematical expression of the lever principle is based on the principle of torque balance, where torque is the result of multiplying the applied force by the lever arm, which is the perpendicular distance between the point of force application and the fulcrum. In an ideal scenario, the condition for a lever system to achieve a state of equilibrium is that the clockwise torque equals the counterclockwise torque. As illustrated in the Fig. 2, F_1 and F_2 represent the forces applied at the two ends of the lever, while L_1 and L_2 denote the distances from the fulcrum to the points where F_1 and F_2 are respectively applied. When the lever is in a balanced state, the following relationship is satisfied:

$$F_1 \cdot L_1 = F_2 \cdot L_2 \quad (19)$$

where, F_1 and F_2 represent the effort and load forces, and L_1 and L_2 represent the effort arm and load arm, respectively. This expression reveals the principle of torque balance, stating that the product of the force and its lever arm on one side equals the product of the force and its lever arm on the other side.

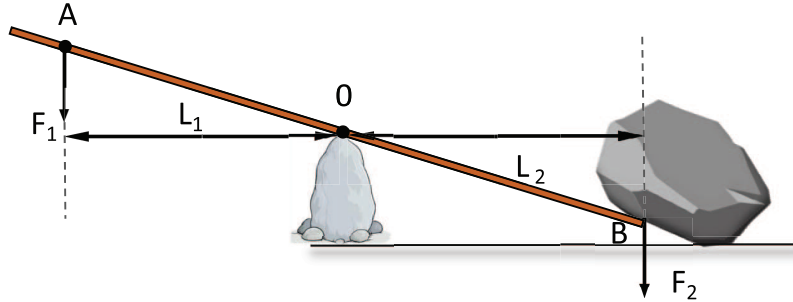


Figure 2: A schematic diagram of the lever principle

In the standard OBL introduced in Section 2.3, the particle's position is simply reversed through axial symmetry to obtain the opposition value. However, if the solution vector itself exhibits axial symmetry, the standard OBL operation will not change the solution, rendering the operation ineffective and increasing redundant computations. To avoid this inefficacy, this paper proposes a lever principle-based OBL (lever-OBL) strategy. To construct the lever-OBL model, we consider the particle's rank ($Rank(i)$) as the effort, the entire population size as the load, and the particle's position (X_i) as the effort arm. This strategy also incorporates Cauchy perturbation using $Cauchy(0,1)$ to prevent operational inefficacy. According to Eq. (19), the new particle position ($\tilde{X}_i$) can be derived as follows in Eq. (20).

$$\tilde{X}_i = \frac{Rank(i)}{NP} \cdot X_i \cdot Cauchy(0,1) \quad (20)$$

3.3 Local Selection and Re-Optimization Strategy

This section introduces a local selection and re-optimization strategy (LSR). In this strategy, a random dimension of the historical best position (X_p^d) of the global best particle (X_p) is chosen. According to the LSR, randomly select 80% of the data from all dimensions as the new X_p . A perturbation is then introduced in that dimension based on a Cauchy distribution using $Cauchy(x_0, \gamma)$. The value of X_p^d is ultimately obtained through Eq. (21), and X_p^d is taken as X_p . The utilization of this LSR aims to enhance the diversity and global exploration of the search process. The pseudocode of the LSR is presented in Algorithm 1.

$$\begin{cases} X_p^d = X_p^d + (x_d^U - x_d^L) \cdot Cauchy(x_0, \gamma), \\ \gamma = 1 - 0.8 \times \frac{iter}{MaxIter}, x_0 = 0. \end{cases} \quad (21)$$

Algorithm 1: The pseudo-code of LSR strategy

Input: GB .

Output: X_p .

1: $X_p = GB$; $d = rand(1, D)$;

2: /*The core of LSR */

3: Randomly select 80% of the data from all dimensions as the new X_p^d using Eq. (21);

4: $X_p = X_p^d$;

5: Calculate the fitness by $Fitness(X_p)$ using Eq. (22);

(Continued)

Algorithm 1 (continued)

```

6: if  $fitness < Fitness(GB)$ 
7:    $GB = X_p$ ;
8: else
9:    $X_p$  is utilized to replace the worst particle;
10: end if

```

3.4 The Proposed Algorithm

In most study, an EC-based FS algorithm is employed as the solution method, where feature selection is represented in binary encoding is used to represent whether a feature is selected, as described in the previous [Section 2.4](#), with $x_j \in \{0, 1\}$. Then, a threshold ($thres = 0.5$) is set. If $x_j > thres$, the feature is considered selected ($x_j = 1$); instead, the feature is not selected ($x_j = 0$). Additionally, in this study, as with most feature selection research papers, the wrapper-based FS method utilizes the KNN algorithm as the evaluation method, with fitness values calculated using [Eq. \(22\)](#).

$$Fitness = \alpha \cdot E_r(S_d) + \beta \cdot \frac{Num_{s.t.}}{D} \quad (22)$$

where, $Num_{s.t.}$ represents the number of features in the selected subset, D denotes the total number of features, and $E_r(S_d)$ is the error rate of the selected feature subset, defined as the ratio of misclassified instances to the total number of instances. The parameters α and β are employed to balance the relative significance of the error rate and the quantity of selected features, with $\alpha + \beta = 1$. In most EC-based FS methods, the primary objective is to improve classification accuracy. Therefore, this study follows the mainstream approach by setting $\alpha = 0.9$ and $\beta = 0.1$ [\[48,53,54\]](#).

The complete pseudocode in Algorithm 2 provides a detailed explanation of the steps in the code execution of the algorithm. At the start of lever-FLPSO, the initial strategy involves randomly initializing NP particles. In line 6, the Euclidean distance (d_g) of particles is computed. Subsequently, in line 8, the particle's state is determined by $\tau_s(\lambda)$, and based on different states, the algorithm selects various acceleration parameters. As outlined in line 9, the entire algorithm employs the computation approach for the nonlinear weight (ω) as depicted in [Eq. \(9\)](#) and adjusts the values of c_1 and c_2 based on the distinct states ($\tau_s(\lambda)$). To filter out redundant features from the optimal solution, LSR is chosen to enhance the diversity of the particle population and use 80% of the optimal solution to replace the optimal solution vector in line 10. Finally, in line 16, the vectors representing the personal best position (PB_i) and the global best position (GB) are updated. Additionally, in line 18, opposition-based learning based on the leverage principle is used to obtain the opposite solution, ensuring that significant feature combinations that contribute meaningfully are not overlooked until all iterations have concluded. To illustrate the adaptive parameter control process of the lever-FLPSO, a flowchart is provided in [Fig. 3](#).

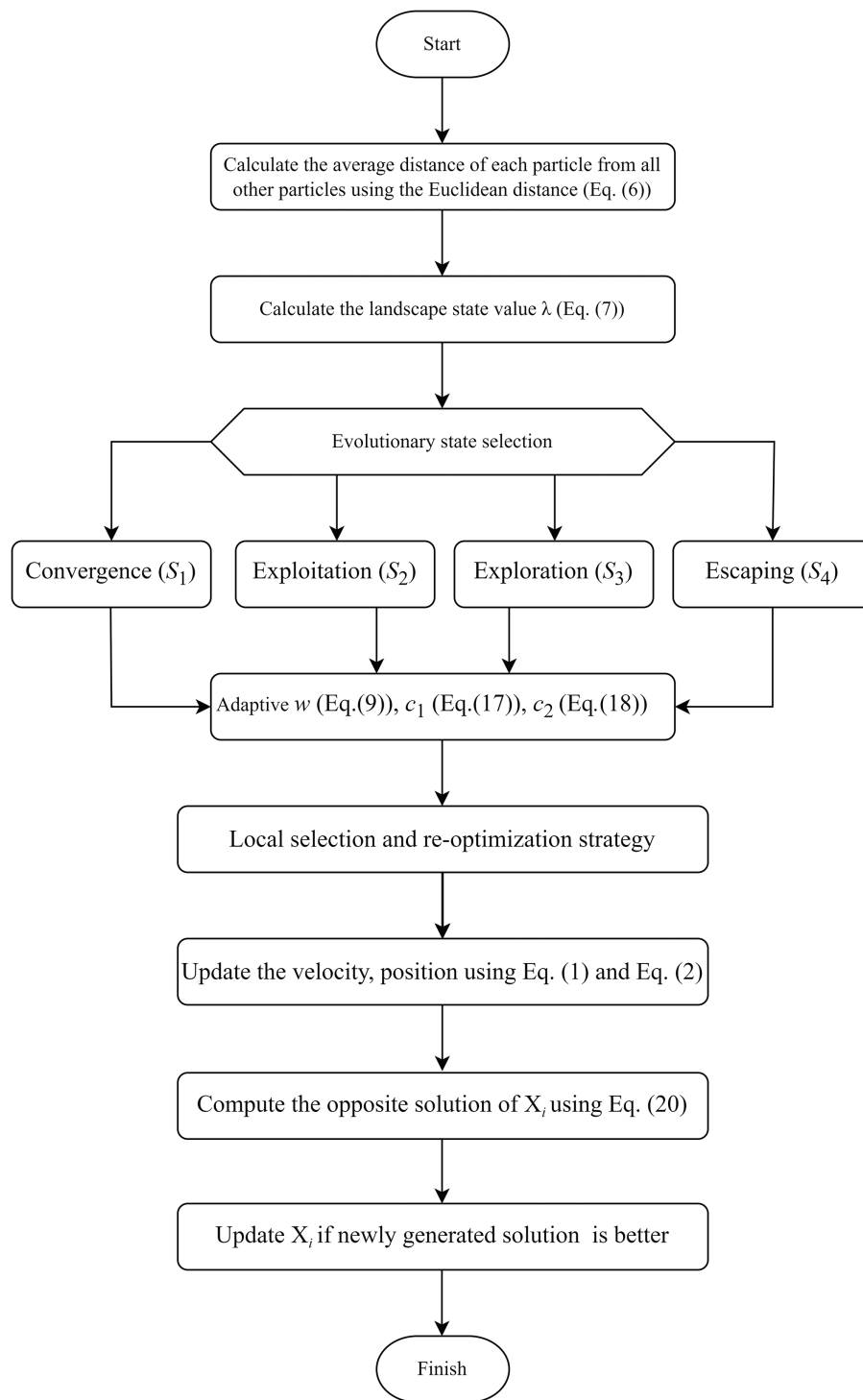


Figure 3: Flowchart of the adaptive parameter control process of the lever-FLPSO

Algorithm 2: The pseudo-code of lever-FLPSO algorithm

Input: Training data, NP, D, MaxIter.
Output: The optimal solution (*Fitness*).

- 1: **for** $i = 1$ to NP **do**
- 2: Randomly initialize v_i, x_i ;
- 3: **end for**
- 4: **while** iter < MaxIter
- 5: Calculate average distance d_g using Eq. (6);
- 6: Calculate the landscape status value λ using Eq. (7);
- 7: Selection of the four states $\tau_s(\lambda)$ using Eq. (8);
- 8: Update the w, c_1 and c_2 using Eqs. (9), (17), (18);
- 9: Update the **GB** using the LSR from Algorithm 1;
- 10: **for** $i = 1$ to NP **do**
- 11: **for** $j = 1$ to D **do**
- 12: Update the velocity, position using Eqs. (1) and (2);
- 13: **end for**
- 14: **end for**
- 15: Update **PB_i**, **GB**;
- 16: Compute the opposite solution of X_i using Eq. (20);
- 17: Update X_i if newly generated solution ($\tilde{X}_i$) is better;
- 18: iter = iter + 1;
- 19: **end while**
- 20: Getting the optimal feature subset;
- 21: Load the test data and evaluate the classification accuracy.

3.5 Algorithm Complexity Analysis

For the PSO algorithm, it is well-known that the time complexity is $O(\text{MaxIter} \times \text{NP} \times D)$. Next, the time complexity will be analysed according to the usual analysis methods [55]. For the lever-FLPSO, according to Eqs. (6)–(8), the complexity of population status measurement is $O(O(\text{NP}^2 \times D) + O(\text{NP}) + O(1))$; for LSR, since only 80% of the GB is processed, its time complexity is $O(0.8 \times D)$; the time complexity of the lever-OBL strategy is $O(\text{NP} \times D)$. Given these components, the time complexity for each iteration is dominated by the most expensive operation, which is the distance calculation ($O(\text{NP}^2 \times D)$). Therefore, the overall time complexity of the algorithm is $O(\text{MaxIter} \times \text{NP}^2 \times D)$. However, the study employed a smaller population size to tackle high-dimensional complex problems, meaning the increase in complexity does not pose a limitation to the improvement of the solving algorithm's quality.

4 Experimental Results and Analysis

In this part of the document, detailed information about the 24 utilized datasets is first presented. The comparative algorithms and specific parameter settings for the experiments are described in conjunction with this section. In the experimental analysis, the performance of the proposed algorithm is evaluated through a comparison with 8 advanced PSO variants and 5 swarm intelligence methods. The results are then presented, followed by a discussion of the findings.

4.1 Dataset

This study selected 10 UCI datasets [56] and 14 high-dimensional ASU datasets [57] for the evaluation of the algorithm's performance. The brief information about the datasets is shown in Table 2. For each dataset, 10-fold cross-validation was used to evaluate the feature subset, with 70% of the instances randomly selected for training and 30% for testing.

Table 2: The brief information for the 24 datasets used in this study

No.	Type	Dataset	Features	Instances	Classes
1	UCI datasets	Glass	9	214	7
2		Wine	13	178	3
3		Heart	13	303	5
4		Zoo	16	101	7
5		Parkinsons	22	195	2
6		Dermatology	34	366	6
7		Ionosphere	34	351	2
8		LungCancer	56	32	3
9		Sonar	60	208	2
10		Musk1	166	476	2
11	High-dimensional datasets	Yale	1024	165	15
12		SRBCT	2308	83	4
13		warpAR10P	2400	130	10
14		warpPIE10P	2420	210	10
15		Lymphoma	5026	62	3
16		TOX-171	5748	171	4
17		9_Tumors	5726	60	9
18		Leukemia	7070	72	2
19		ALLAML	7129	72	2
20		nci9	9712	60	9
21		Adenocarcinoma	9868	76	2
22		Prostate_Tumor	10509	102	2
23		11_Tumors	12533	174	11
24		Lung_Cancer	12600	203	5

4.2 Experimental Parameter Settings

Considering the unique characteristics of lever-FLPSO, comparisons are conducted with 8 sophisticated PSO approaches applied in FS: TLPSO [44], COPSO [45], ASRPPSO [37], PLTVACIW-PSO [38], SFE-PSO [47], EBPSO1 [46] and EBPSO2 [46], and MEL-PSO [48]. These excellent PSO variants obviously exhibit the characteristics of population adaptive partitioning, based on OBL, based on Euclidean distance control of acceleration parameters, linear variation of acceleration parameters, etc., which are more appropriate for experimental comparison with the lever-FLPSO proposed in this paper. Table 3 provides detailed descriptions of the parameters used for both the comparison algorithms and the proposed algorithm in this study. Additionally, it was compared with five other heuristic algorithms with the above characteristics: BASO [1], ECSA [23], SSA [18], ISSA [20] and mgFPA [21]. To ensure fairness in the experiments, all

algorithm parameters were set identically as follows. Moreover, all simulation environments in this study were conducted using Windows 11, 12th Gen Intel(R) Core(TM) i9-12900K 3.20 GHz, and Python 3.11. As the work has not yet been published, the code will be made available on an open-source platform in the future. The following are the definitions of the parameters. Should there be any questions regarding the parameter settings, please feel free to contact the corresponding author to request the code.

- Population size (NP): 10
- The problem dimension (D): the number of features in the dataset
- Maximum number of algorithm iterations ($MaxIter$): 100
- The independent number of runs (Run_Num): 20
- The number of nearest neighbors (k) in the K -nearest neighbor: 5

Table 3: Basic parameter settings for comparative algorithms

Algorithm	Year	Parameter setting
mgFPA [21]	2019	$P = 0.8, \gamma = 0.01, \beta = 1.5.$
BASO [1]	2020	$\alpha = 50, \beta = 0.2.$
ECSA [23]	2020	$AP = 0.1, FL = 1.5.$
SSA [18]	2020	$c_1 = 2 \times \exp(-(4 \times iter/MaxIter)^2), c_2 = c_3 = rand().$
ISSA [20]	2020	$c_1 = 2 \times \exp(-(4 \times iter/MaxIter)^2), c_2 = c_3 = rand().$
TLPSO [44]	2021	$\phi = 0.4, \delta = 0.3 \sim 0.1.$
COPSO [45]	2022	$w = 0.5 + 0.5 \cdot rand() \text{ or } w = 0.5 \cdot rand(),$ $c_1 = 2, c_2 = 2.$
ASRPPSO [37]	2022	$w = 0.9 \sim 0.4, a = -0.035, b = -0.275,$ $m = 0, n = 1.2.$
PLTVACIW-PSO [38]	2023	$w = 0.9 \sim 0.4, c_1 = 1.28 \sim 1.05, c_2 = 1.05 \sim 1.28.$
SFE-PSO [47]	2023	$w = 1, c_1 = 1.5, c_2 = 2, UR = 0.3 \sim 0.1.$
EBPSO1 [46]	2024	$w = 0.5, c_1 = 0.5, c_2 = 1, u = 0.9, l = 0.$
EBPSO2 [46]	2024	$w = 0.5, c_1 = 0.5, c_2 = 1, \tau = 0.5.$
MEL-PSO [48]	2024	$w = 0.9, c_1 = 2, c_2 = 2, c_3 = 2.$
Lever-FLPSO	–	show in Eqs. (9), (17) and (18).

4.3 Experimental Results and Analysis

In this experiment, a range of evaluation metrics were calculated to evaluate the performance of lever-FLPSO. These metrics include the average fitness value, average classification accuracy, and the number of features selected. Moreover, convergence plots, box plots, and spider diagrams were utilized to offer a more intuitive representation of the experimental outcomes. The most favorable outcomes of several algorithms in the experiment are highlighted in bold.

4.3.1 Experimental Results

The convergence curves of the experimental results are presented in Fig. 4. It is apparent that, in comparison to other PSO variants and heuristic algorithms, lever-FLPSO demonstrates a more rapid convergence towards the optimal solution. Additionally, lever-FLPSO achieves faster and better convergence on high-dimensional datasets (e.g., “SRBCT”, “warpPIE10P”, “TOX-171” and “Leukemia”). This is primarily due to the FL strategy, which strikes a proper balance between exploration and exploitation. Furthermore,

in Fig. 5, by using LSR to continuously reduce the number of features in the optimal solution, the convergence speed is enhanced. Compared to ISSA, lever-FLPSO tends to search in more promising regions, attributed to the lever-OBL providing high-quality solutions and improving the algorithm's convergence.

Tables 4 and 5 list the average classification accuracy and the standard deviation of fitness values. The lever-FLPSO consistently achieves the top rank among the compared algorithms, demonstrating particularly outstanding performance on high-dimensional datasets. These results demonstrate the superior performance of lever-FLPSO in searching for global optimal solutions in complex problems. The primary advantage of the method is attributed to the introduction of the lever-OBL scheme, which enhances the algorithm's ability to identify optimal feature combinations in high-dimensional problems. It is worth noting that, while lever-FLPSO does not consistently attain the highest classification accuracy on certain datasets, the difference from the top-performing methods is minimal. Furthermore, lever-FLPSO maintains a consistent average ranking of first place across the 24 datasets. In terms of robustness and consistency, lever-FLPSO typically provides highly consistent results due to the lower Std values. The implemented FL scheme significantly improves the algorithm's stability during each search, enabling it to evolve more effectively towards the optimal solution based on the state of the population.

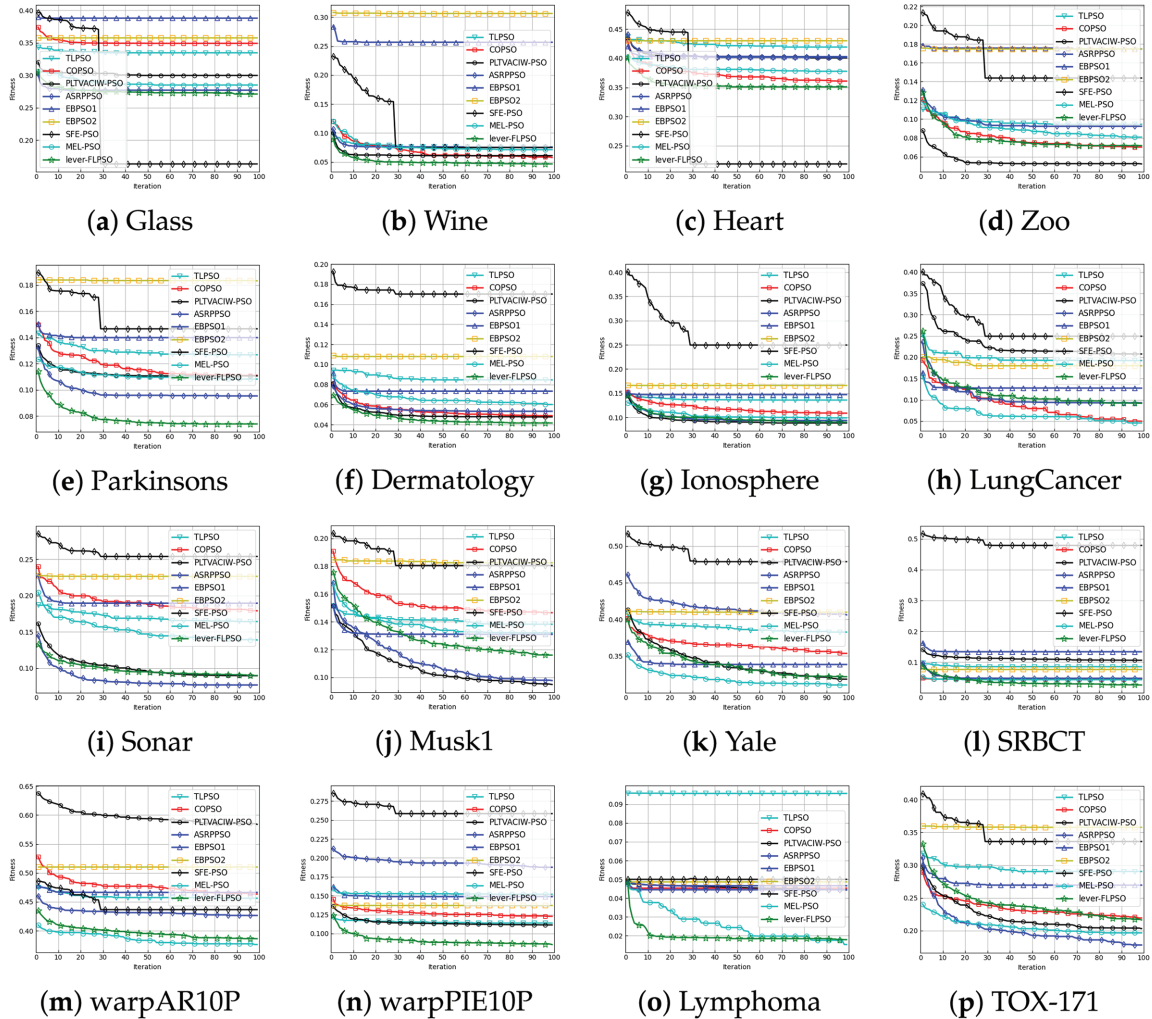


Figure 4: (Continued)

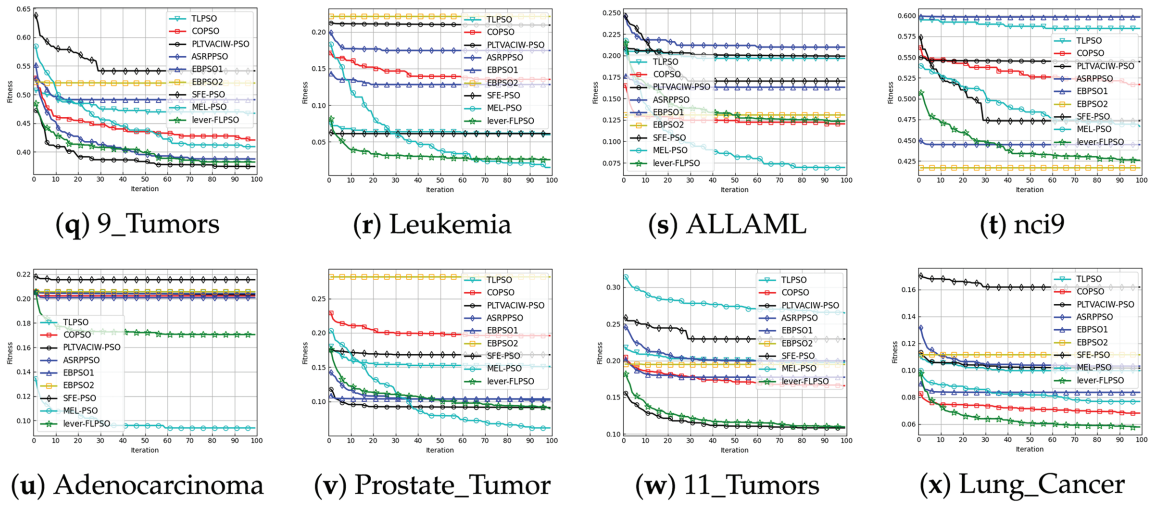


Figure 4: The convergence curves of 20 independent runs for lever-FLPSO and and 6 swarm intelligence algorithms

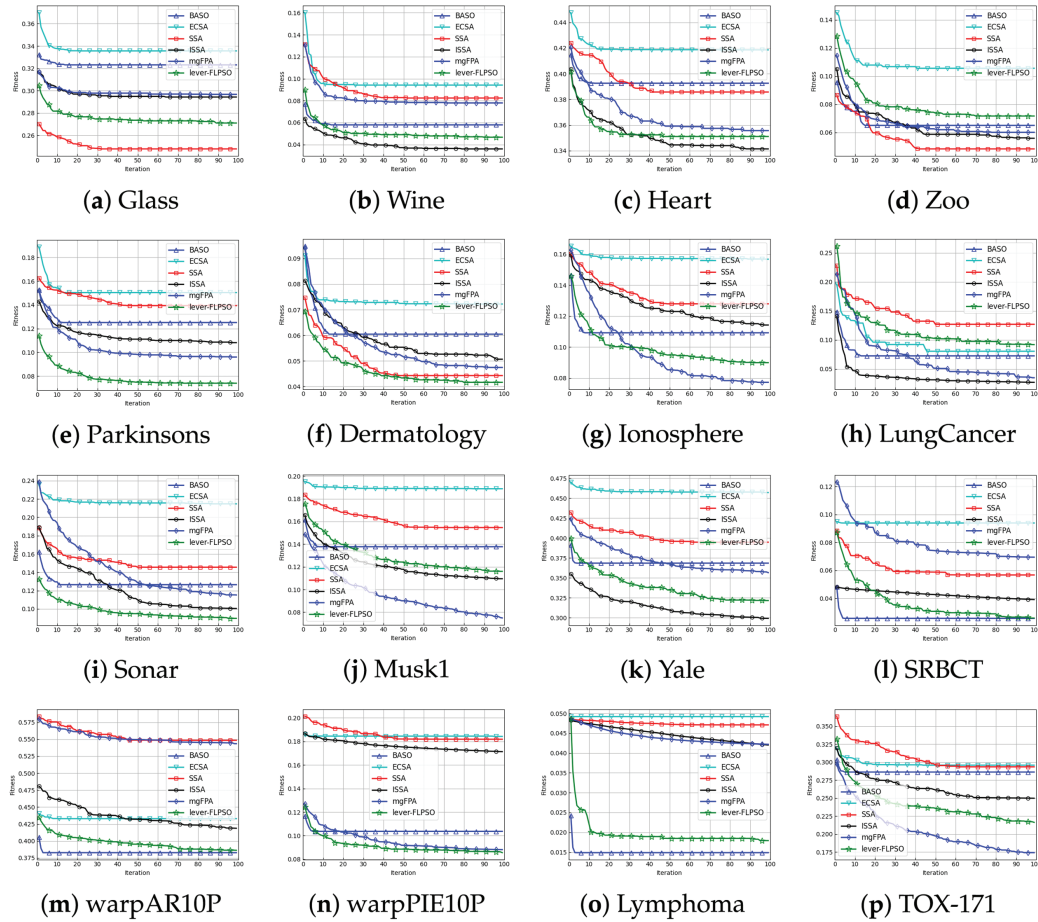


Figure 5: (Continued)

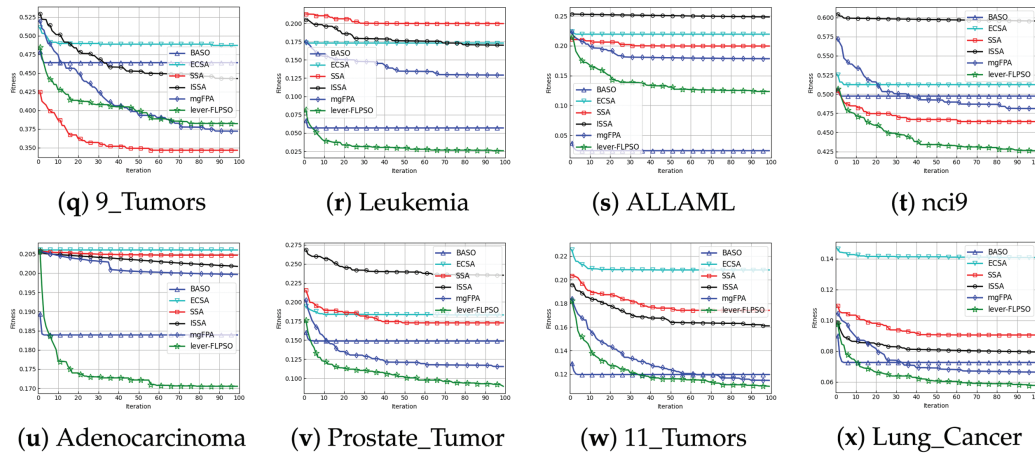


Figure 5: The convergence curves of 20 independent runs for lever-FLPSO and 5 swarm intelligence algorithms

Table 4: The comparative results of classification accuracy between lever-FLPSO and advanced PSO algorithms on the datasets

Name	Type	TLPSO	COPSO	ASRPPSO	PLTVACIW_PSO	SFE-PSO	EBPSO1	EBPSO2	MEL-PSO	Lever-FLPSO
Glass	Mean	0.6777	0.6615	0.7485	0.7054	0.6508	0.6523	0.6692	0.7354	0.7323
	Std.	0.0191	0.0000	0.0196	0.0177	0.0213	0.0185	0.0277	0.0226	0.0102
Wine	Mean	0.9611	0.9769	0.9519	0.9694	0.9009	0.7611	0.7194	0.9583	0.9815
	Std.	0.0202	0.0000	0.0444	0.0121	0.0663	0.0777	0.0559	0.0254	0.0000
Heart	Mean	0.5794	0.6383	0.5861	0.5856	0.5850	0.5844	0.5661	0.6006	0.6411
	Std.	0.0266	0.0108	0.0110	0.0294	0.0252	0.0163	0.0224	0.0237	0.0205
Zoo	Mean	0.9419	0.9710	0.9435	0.9823	0.8661	0.8371	0.8726	0.9565	0.9661
	Std.	0.0281	0.0174	0.0247	0.0160	0.0588	0.0070	0.0260	0.0211	0.0070
Parkinsons	Mean	0.9034	0.9229	0.9178	0.8983	0.8822	0.8847	0.8551	0.9025	0.9424
	Std.	0.0121	0.0125	0.0280	0.0093	0.0377	0.0198	0.0125	0.0118	0.0083
Dermatology	Mean	0.9597	0.9949	0.9856	0.9907	0.8792	0.9787	0.9519	0.9773	0.9949
	Std.	0.0158	0.0062	0.0069	0.0093	0.0672	0.0161	0.0454	0.0085	0.0069
Ionosphere	Mean	0.8943	0.9151	0.9264	0.9330	0.9080	0.9000	0.8811	0.9009	0.9288
	Std.	0.0129	0.0103	0.0125	0.0130	0.0211	0.0249	0.0210	0.0199	0.0125
LungCancer	Mean	0.8333	0.9944	0.9333	0.8000	0.7778	0.9056	0.8722	0.9833	0.9333
	Std.	0.0745	0.0000	0.0544	0.0831	0.1111	0.0636	0.0530	0.0397	0.0544
Sonar	Mean	0.8690	0.8540	0.9587	0.9437	0.8016	0.8381	0.8341	0.8786	0.9468
	Std.	0.0206	0.0108	0.0154	0.0191	0.0216	0.0198	0.0155	0.0232	0.0115
Muskl	Mean	0.8997	0.8902	0.9420	0.9430	0.8657	0.9052	0.8608	0.9024	0.9189
	Std.	0.0131	0.0106	0.0152	0.0109	0.0248	0.0162	0.0204	0.0139	0.0170
Yale	Mean	0.6290	0.6630	0.5990	0.6980	0.5420	0.6770	0.6070	0.7030	0.6790
	Std.	0.0232	0.0114	0.0148	0.0244	0.0303	0.0182	0.0182	0.0222	0.0184
SRBCT	Mean	0.9600	1.0000	0.9980	0.9300	0.9440	0.9040	0.9700	1.0000	0.9980
	Std.	0.0219	0.0000	0.0087	0.0214	0.0294	0.0294	0.0249	0.0000	0.0087
warpAR10P	Mean	0.5474	0.5410	0.5769	0.4013	0.5923	0.5346	0.4872	0.6244	0.6090
	Std.	0.0260	0.0160	0.0128	0.0295	0.0519	0.0168	0.0000	0.0295	0.0160
warpPIE10P	Mean	0.9286	0.9198	0.8437	0.9254	0.7873	0.8881	0.9024	0.8762	0.9349
	Std.	0.0094	0.0035	0.0126	0.0102	0.0272	0.0117	0.0057	0.0081	0.0111
Lymphoma	Mean	0.9474	0.9998	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
	Std.	0.0000	0.0000	0.0000	0.0000	0.0531	0.0000	0.0000	0.0513	0.0000
TOX-171	Mean	0.7327	0.8135	0.8567	0.8279	0.6990	0.7548	0.6663	0.8240	0.8038
	Std.	0.0210	0.0162	0.0352	0.0392	0.0000	0.0278	0.0204	0.0297	0.0308
9_Tumors	Mean	0.5361	0.5889	0.6222	0.6361	0.4722	0.5083	0.4944	0.5472	0.6194
	Std.	0.0403	0.0272	0.0624	0.0541	0.0000	0.0640	0.0299	0.0000	0.0318
Leukemia	Mean	0.9886	0.9045	0.8568	0.8182	1.0000	0.9114	0.8091	0.9841	0.9977
	Std.	0.0197	0.0136	0.0162	0.0000	0.0735	0.0304	0.0308	0.0709	0.0099
ALLAML	Mean	0.8364	0.9205	0.8182	0.8295	0.9091	0.8727	0.9091	0.9227	0.8955
	Std.	0.0223	0.0000	0.0000	0.0197	0.0518	0.0182	0.0000	0.0500	0.0208
nci9	Mean	0.4056	0.4806	0.5556	0.4472	0.5528	0.3889	0.5917	0.5111	0.5583
	Std.	0.0255	0.0265	0.0000	0.0121	0.0541	0.0000	0.0318	0.0598	0.0276

(Continued)

Table 4 (continued)

Name	Type	TLPSO	COPSO	ASRPPSO	PLTVACIW _PSO	SFE- PSO	EBPSO1	EBPSO2	MEL- PSO	Lever- FLPSO
Adenocarcinoma	Mean	0.8261	0.8261	0.8261	0.8261	0.8261	0.8261	0.8261	0.8957	0.8304
	Std.	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0288	0.0130
Prostate_Tumor	Mean	0.8871	0.8371	0.9387	0.9516	0.8919	0.9387	0.7419	0.9323	0.9387
	Std.	0.0239	0.0070	0.0141	0.0161	0.0185	0.0141	0.0306	0.0268	0.0248
11_Tumors	Mean	0.8358	0.8726	0.8321	0.9330	0.8113	0.8575	0.8415	0.7585	0.9217
	Std.	0.0261	0.0082	0.0132	0.0152	0.0215	0.0174	0.0125	0.0194	0.0217
Lung_Cancer	Mean	0.9443	0.9795	0.9385	0.9402	0.8893	0.9615	0.9311	0.9656	0.9672
	Std.	0.0096	0.0071	0.0114	0.0094	0.0186	0.0094	0.0084	0.0102	0.0090
Mean rank		5.83	3.96	4.04	3.96	6.58	5.83	6.71	3.71	2.38
Final rank		6	3	5	3	8	6	9	2	1

Note: The comparative results of classification accuracy between lever-FLPSO and advanced PSO algorithms on the datasets, and the first-best is highlighted in bold.

Table 5: The comparative results of classification accuracy between lever-FLPSO and swarm intelligence algorithms on the datasets

Name	Type	BASO	ECSA	SSA	ISSA	mgFPA	Lever- FLPSO
Glass	Mean	0.6154	0.7031	0.7785	0.7231	0.7254	0.7323
	Std.	0.0000	0.0110	0.0191	0.0000	0.0122	0.0102
Wine	Mean	0.7037	0.9620	0.9546	0.9963	0.9528	0.9815
	Std.	0.0000	0.0040	0.0109	0.0074	0.0109	0.0000
Heart	Mean	0.4667	0.5956	0.6056	0.6489	0.6356	0.6411
	Std.	0.0000	0.0147	0.0167	0.0096	0.0213	0.0205
Zoo	Mean	0.7742	0.9500	0.9903	0.9823	0.9710	0.9661
	Std.	0.0000	0.0279	0.0148	0.0160	0.0097	0.0070
Parkinsons	Mean	0.7966	0.8983	0.8805	0.8983	0.9110	0.9424
	Std.	0.0000	0.0000	0.0037	0.0000	0.0220	0.0083
Dermatology	Mean	0.8333	0.9921	0.9949	0.9852	0.9829	0.9949
	Std.	0.0000	0.0084	0.0069	0.0061	0.0061	0.0069
Ionosphere	Mean	0.8113	0.8943	0.8953	0.9033	0.9349	0.9288
	Std.	0.0000	0.0092	0.0115	0.0119	0.0111	0.0125
LungCancer	Mean	0.4444	1.0000	0.9056	1.0000	0.9889	0.9333
	Std.	0.0000	0.0000	0.0636	0.0000	0.0333	0.0544
Sonar	Mean	0.7460	0.8397	0.8873	0.9278	0.9103	0.9468
	Std.	0.0000	0.0187	0.0173	0.0248	0.0190	0.0115
Musk1	Mean	0.7902	0.8654	0.8794	0.9266	0.9654	0.9189
	Std.	0.0000	0.0117	0.0136	0.0130	0.0102	0.0170
Yale	Mean	0.5200	0.5740	0.6150	0.7110	0.6520	0.6790
	Std.	0.0000	0.0169	0.0199	0.0173	0.0133	0.0184
SRBCT	Mean	0.8800	0.9600	0.9900	1.0000	0.9700	0.9980
	Std.	0.0000	0.0000	0.0173	0.0000	0.0249	0.0087
warpAR10P	Mean	0.5641	0.5897	0.4449	0.5821	0.4436	0.6090
	Std.	0.0000	0.0000	0.0168	0.0325	0.0164	0.0160
warpPIE10P	Mean	0.8571	0.8603	0.8516	0.8532	0.9508	0.9349

(Continued)

Table 5 (continued)

Name	Type	BASO	ECSA	SSA	ISSA	mgFPA	Lever-FLPSO
Lymphoma	Std.	0.0000	0.0063	0.0091	0.0069	0.0099	0.0111
	Mean	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
TOX-171	Mean	0.5385	0.7577	0.7288	0.7731	0.8606	0.8038
	Std.	0.0000	0.0128	0.0201	0.0240	0.0271	0.0308
9_Tumors	Mean	0.4444	0.5333	0.6694	0.5583	0.6389	0.6194
	Std.	0.0000	0.0369	0.0327	0.0411	0.0541	0.0318
Leukemia	Mean	0.9545	0.8636	0.8318	0.8614	0.9068	0.9977
	Std.	0.0000	0.0000	0.0208	0.0226	0.0336	0.0099
ALLAML	Mean	0.9091	0.8182	0.8318	0.7727	0.8523	0.8955
	Std.	0.0000	0.0000	0.0208	0.0000	0.0197	0.0208
nci9	Mean	0.3889	0.4972	0.5389	0.3889	0.5167	0.5583
	Std.	0.0000	0.0121	0.0309	0.0000	0.0500	0.0276
Adenocarcinoma	Mean	0.8261	0.8261	0.8261	0.8261	0.8283	0.8304
	Std.	0.0000	0.0000	0.0000	0.0000	0.0095	0.0130
Prostate_Tumor	Mean	0.8065	0.8694	0.8629	0.7903	0.9242	0.9387
	Std.	0.0000	0.0160	0.0225	0.0191	0.0154	0.0248
11_Tumors	Mean	0.8868	0.8500	0.8613	0.8736	0.9264	0.9217
	Std.	0.0000	0.0094	0.0172	0.0170	0.0157	0.0217
Lung_Cancer	Mean	0.9016	0.9172	0.9541	0.9639	0.9787	0.9672
	Std.	0.0000	0.0097	0.0111	0.0098	0.0105	0.0090
Mean rank		4.92	3.96	3.58	3.04	2.58	1.92
Final rank		6	5	4	2	3	1

Note: The comparative results of classification accuracy between lever-FLPSO and swarm intelligence algorithms on the datasets, and the first-best is highlighted in bold.

Tables 6 and 7 display the results concerning the number of selected features. Notably, lever-FLPSO achieves the minimal average number of selected features on most datasets, with only a few exceptions. Among the 13 compared algorithms across the 24 datasets, its average rankings are **2.38** and **1.92**, respectively, with an overall rank of first. The primary reason for lever-FLPSO's improved efficiency is the use of LSR to reduce the number of features in the optimal solution when addressing the FS problem, thereby selecting the best and significantly enhancing the selection efficiency. Furthermore, the FL strategy underscores a balance between exploration and exploitation in the search process, thereby enhancing the effectiveness of identifying high accuracy alignments with the fewest possible features.

Table 6: The comparative results of the number of selected features between lever-FLPSO and advanced PSO algorithms on the datasets

Name	TLPSO	COPSO	ASRPPSO	PLTVACIW- PSO	SFE-PSO	EBPSO1	EBPSO2	MEL- PSO	Lever- FLPSO
Glass	4.00	4.00	4.55	3.10	3.95	3.00	2.40	4.20	2.70
Wine	5.30	4.80	4.20	4.35	4.55	3.75	4.85	4.40	3.90

(Continued)

Table 6 (continued)

Name	TLPSO	COPSO	ASRPPSO	PLTVACIW_ PSO	SFE-PSO	EBPSO1	EBPSO2	MEL- PSO	Lever- FLPSO
Heart	5.30	4.60	3.80	3.60	3.85	3.75	5.15	2.80	3.20
Zoo	6.75	7.05	6.65	5.80	5.9	1.40	3.00	6.65	6.20
Parkinsons	8.75	9.15	4.70	4.20	7.8	3.25	4.75	4.50	3.30
Dermatology	16.45	15.10	13.75	13.40	11.75	7.05	8.40	13.50	12.60
Ionosphere	13.85	11.00	9.30	9.55	6.25	2.85	2.95	3.35	7.30
LungCancer	23.70	25.10	18.40	15.55	12.15	23.80	36.40	20.35	17.25
Sonar	27.75	28.50	23.70	23.25	12.85	26.25	46.30	17.65	22.10
Musk1	79.55	79.20	75.50	72.35	49.85	75.85	95.00	73.30	63.25
Yale	498.05	513.90	463.15	475.80	167.25	488.00	574.45	440.85	335.35
SRBCT	1133.60	1035.90	1054.85	995.15	436.85	1089.75	1141.90	988.35	517.80
warpAR10P	1175.40	1208.35	1097.25	1084.80	323.95	1140.30	1175.15	913.95	693.50
warpPIE10P	1202.35	1226.45	1134.20	1071.10	573.6	1163.85	1204.95	980.50	593.90
Lymphoma	1947.30	1825.15	1790.55	1788.00	2011.95	1877.25	1961.85	616.95	721.10
TOX-171	2856.65	2951.30	2818.75	2765.15	572.4	2805.60	3303.80	2683.30	2285.90
9_Tumors	2856.10	2903.15	2729.45	2689.60	939.65	2778.50	3726.50	97.75	2279.20
Leukemia	3505.35	3491.60	3242.75	3232.75	2762.8	3425.85	3521.70	54.00	1492.60
ALLAML	3522.20	3479.95	3288.70	3271.30	554.4	3458.25	3515.45	66.35	2094.30
nci9	4804.55	4811.95	4347.50	4568.65	745.9	4673.90	4836.70	2604.90	2770.15
Adenocarcinoma	4739.05	4425.05	4277.80	4513.20	3967.5	4605.50	4769.30	39.05	1730.60
Prostate_Tumor	5189.60	5181.45	4937.05	4997.80	2983.35	5101.15	5247.00	41.20	3598.00
11_Tumors	6228.30	6403.65	6075.60	6015.70	3640.6	6157.70	6520.15	5949.30	4917.80
Lung_Cancer	6238.30	6256.75	6012.00	6009.15	4469.3	6152.30	6253.80	5766.50	3563.70
Mean rank	7.67	7.50	5.08	4.08	2.83	4.83	7.33	3.04	2.54
Final rank	9	8	6	4	2	5	7	3	1

Note: The comparative results of the number of selected features between lever-FLPSO and advanced PSO algorithms on the datasets, and the first-best is highlighted in bold.

Table 7: The comparative results of the number of selected features between lever-FLPSO and swarm intelligence algorithms on the datasets

Name	BASO	ECSA	SSA	ISSA	mgFPA	Lever-FLPSO
Glass	9.00	6.15	4.35	4.05	4.45	2.70
Wine	13.00	7.80	5.40	4.25	4.60	3.90
Heart	13.00	7.10	4.00	3.30	3.60	3.20
Zoo	16.00	9.70	6.35	6.40	5.45	6.20
Parkinsons	22.00	12.95	7.00	3.65	3.50	3.30
Dermatology	34.00	22.15	13.50	12.70	10.90	12.60
Ionosphere	34.00	20.95	11.45	9.25	6.35	7.30
LungCancer	56.00	45.05	23.45	15.05	13.90	17.25
Sonar	60.00	42.25	26.45	21.30	20.70	22.10
Musk1	166.00	112.45	76.45	72.05	72.60	63.25
Yale	1024.00	754.90	495.15	401.35	445.50	335.35
SRBCT	2308.00	1335.05	1100.60	902.25	978.60	517.80
warpAR10P	2400.00	1528.35	1171.50	1022.45	1024.15	693.50
warpPIE10P	2420.00	1417.15	1167.95	947.85	1060.95	593.90
Lymphoma	4026.00	1983.75	1895.95	1691.75	1698.75	721.10

(Continued)

Table 7 (continued)

Name	BASO	ECSA	SSA	ISSA	mgFPA	Lever-FLPSO
TOX-171	5748.00	4472.15	2846.60	2625.15	2797.15	2285.90
9_Tumors	5726.00	3841.00	2806.55	2585.45	2697.05	2279.20
Leukemia	7070.00	3552.20	3407.40	3199.15	3203.15	1492.60
ALLAML	7129.00	3986.15	3441.45	3128.70	3239.05	2094.30
nci9	9712.00	5804.60	4763.70	4410.70	4474.90	2770.15
Adenocarcinoma	9672.00	4797.50	4656.65	4374.55	4361.25	1730.60
Prostate_Tumor	10509.00	6881.20	5171.15	4883.90	4954.55	3598.00
11_Tumors	12533.00	9192.10	6178.55	5909.55	6080.00	4917.80
Lung_Cancer	12600.00	8356.75	6201.15	5928.00	5954.95	3563.70
Mean rank	6.00	5.00	3.92	2.25	2.54	1.29
Final rank	6	5	4	2	3	1

Note: The comparative results of the number of selected features between lever-FLPSO and swarm intelligence algorithms on the datasets, and the first-best is highlighted in bold.

The Wilcoxon-test, as a non-parametric statistical method, is widely employed for comparative analysis of paired or correlated samples. In this study, a significance level of 0.05 was adopted to evaluate performance differences between lever-FLPSO and several comparison algorithms. Relevant results are documented in Table 8. Here, R^+ and R^- denote the rank sums indicating superior or inferior performance of lever-FLPSO relative to the respective comparison algorithm. If the obtained p -value is less than 0.05, a statistically significant difference between lever-FLPSO and the corresponding algorithm is deemed to exist, denoted as “Yes” in the Significance (Sig.) column; otherwise, the difference is considered insignificant. The specific calculation formula can be found in reference [53].

Table 8: Wilcoxon-test results for comparison algorithms on 24 datasets

Lever-FLPSO vs.	Accuracy				Number			
	R^+	R^-	p -value	Sig.	R^+	R^-	p -value	Sig.
TLPSO	300	0	0.000018	Yes	300	0	0.000018	Yes
COPSO	223	53	0.009730	Yes	300	0	0.000018	Yes
ASRPPSO	171	39	0.013741	Yes	300	0	0.000018	Yes
PLTVACIW_PSO	204.5	71.5	0.043091	Yes	290	10	0.000063	Yes
SFE-PSO	271	5	0.000052	Yes	206	94	0.109580	No
EBPSO1	253	0	0.000040	Yes	274	24	0.000318	Yes
EBPSO2	269	7	0.000068	Yes	281	19	0.000182	Yes
MEL-PSO	191	85	0.106964	No	152	148	0.954431	No
BASO	274	2	0.000035	Yes	300	0	0.000018	Yes
ECSA	261	15	0.000183	Yes	300	0	0.000018	Yes
SSA	225	28	0.001384	Yes	300	0	0.000018	Yes
ISSA	217	59	0.016271	Yes	285	15	0.000115	Yes
mgFPA	178	98	0.223757	No	269	31	0.000674	No

Table 8 presents the results of the Wilcoxon-test comparing lever-FLPSO with the comparison algorithms in terms of classification accuracy and the number of selected features. The data indicates that lever-FLPSO exhibits significant differences from the comparison algorithms in most cases, particularly regarding the number of features, further validating its advantage in feature selection.

To facilitate a clearer understanding of the algorithm's performance, box plots from **Figs. 6** and **7** were used to illustrate the experimental results. In these box plots, the bottom horizontal line represents the minimum value within the dataset, while the top horizontal line indicates the maximum value. The size of the box indicates the dispersion or spread of the results, and the vertical line within the box represents the median. From **Figs. 6** and **7**, the lever-FLPSO achieves the best results for the minimum values on the datasets, with the median also being consistently leading. The data presented in the box plots indicates that the FL strategy effectively identifies the relationships between features in high-dimensional data during each code run, thereby aggregating relevant information and yielding better results.

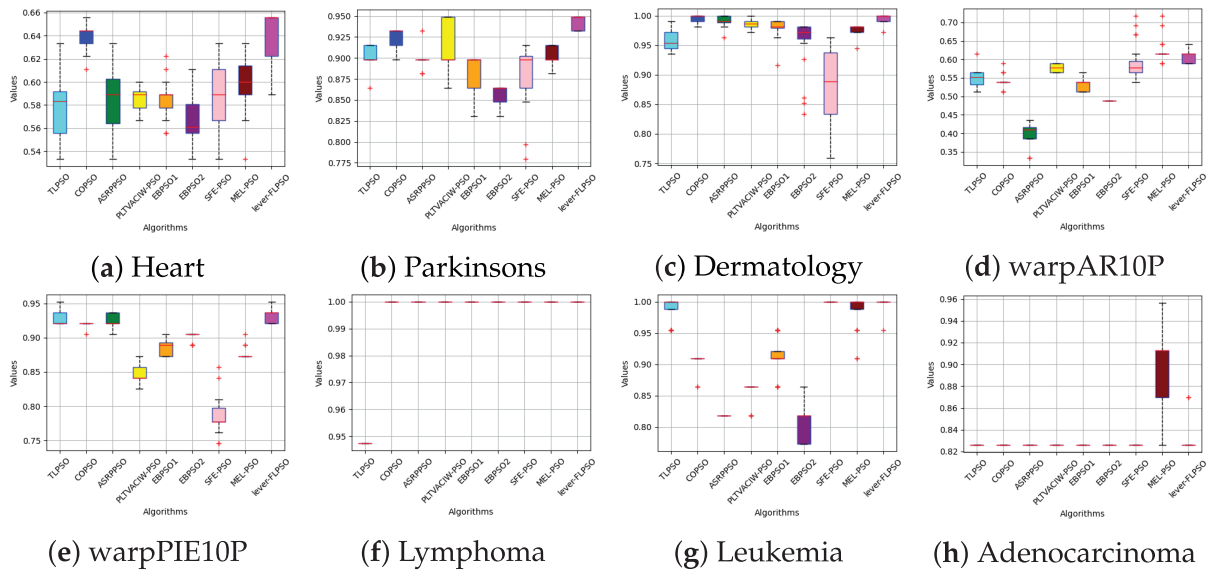


Figure 6: Box plot of accuracy between lever-FLPSO and advanced PSO algorithms on the datasets

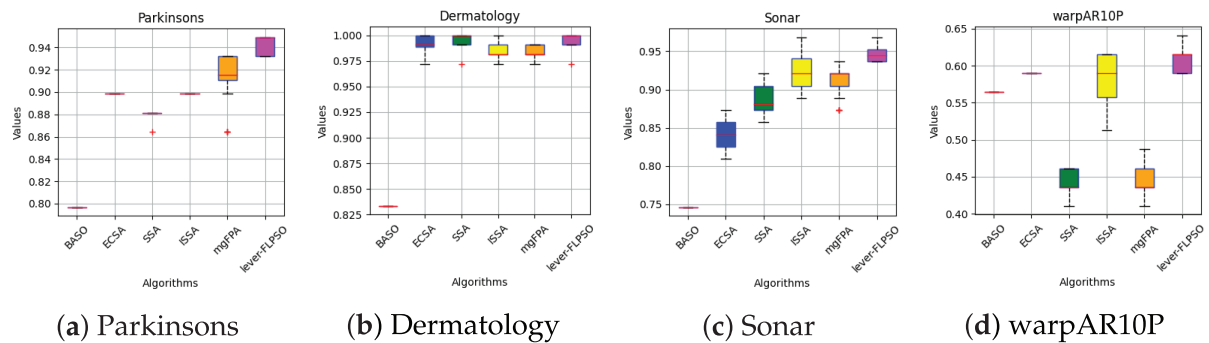


Figure 7: (Continued)

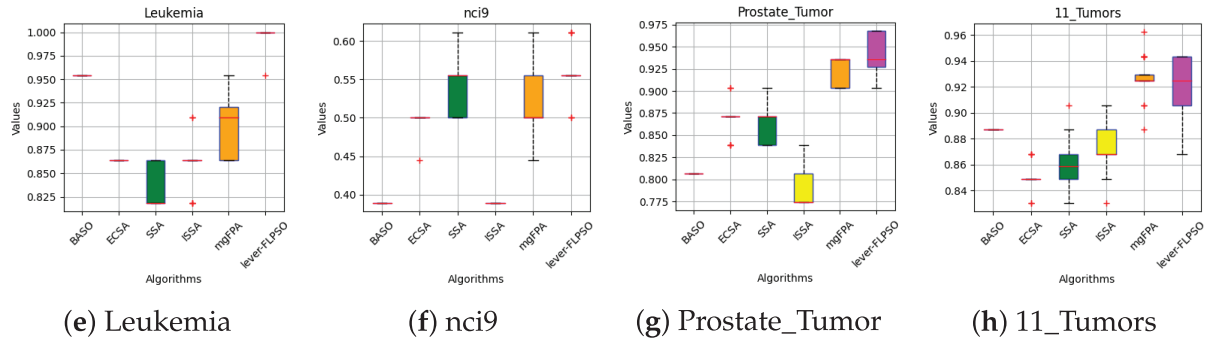


Figure 7: Box plot of accuracy between lever-FLPSO and swarm intelligence algorithms on the datasets

To provide a more intuitive comparison of the rankings of the proposed lever-FLPSO against advanced comparative algorithms across various datasets, spider plots 8 and 9 illustrate the ranking performance of all experimental algorithms on the datasets. It should be noted that in the diagrams, the axes represent the datasets, while the closed lines indicate the algorithms. It is evident that lever-FLPSO is positioned centrally within the spider plot, indicating that it ranks favorably in both classification accuracy and the number of selected features. In Figs. 8a and 9a, compared with the algorithms, lever-FLPSO consistently ranks among the top in classification accuracy across most datasets (e.g., “Wine”, “Heart”, “warpPIE10P”, “Lymphoma”, “Leukemia”, “Lung_Cancer”). Figs. 8b and 9b further highlights its strong performance, where the number of selected features ranks favorably on more than half of the datasets.

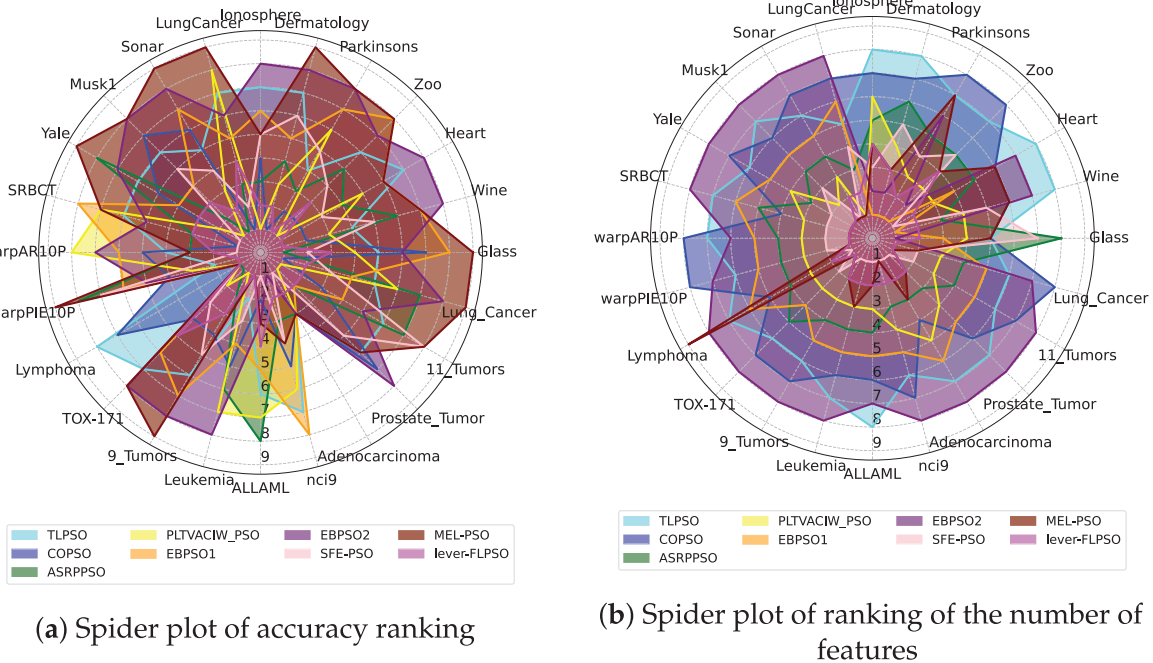
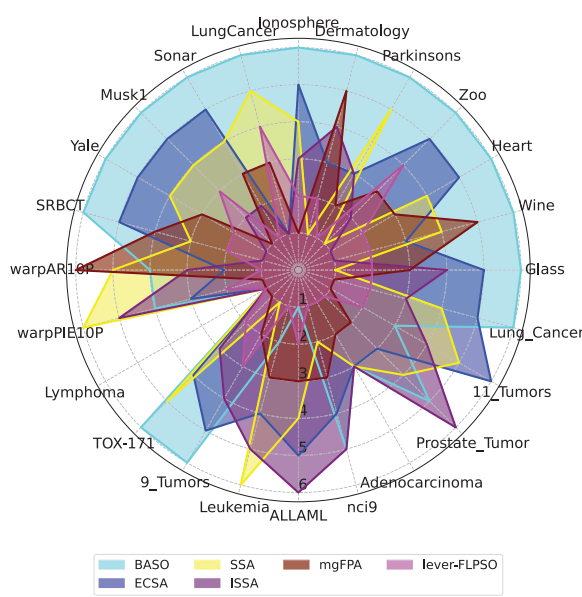
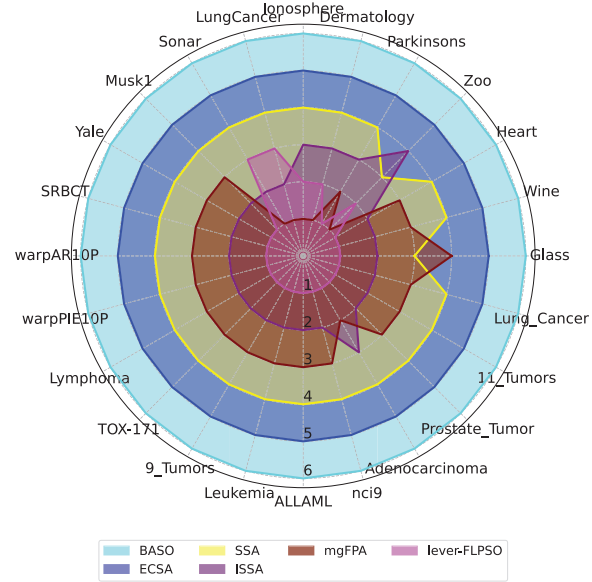


Figure 8: The spider plots between lever-FLPSO and advanced PSO algorithms on the datasets



(a) Spider plot of accuracy ranking



(b) Spider plot of ranking of the number of features

Figure 9: The spider plots between lever-FLPSO and swarm intelligence algorithms on the datasets

The specific execution times (in seconds) of the algorithm across 24 datasets are detailed in [Table 9](#). Feature selection constitutes a practical problem and forms part of the preliminary stages of data processing. The improved feature subsets identified through algorithmic screening can significantly aid subsequent data applications, rendering the extended computation time acceptable. Furthermore, the experimental results presented in the [Tables 4–7](#)—regarding classification accuracy, the number of selected features, and algorithmic execution time—validate that our algorithm possesses competitive capabilities.

Table 9: Average runtime (s) of the algorithm across 24 datasets

Name	BASO	ECSCA	SSA	ISSA	mgFPA	TLPSO	COPSO	ASRPPSO	PLTVA CIW_PSO	SFE- PSO	EBPSO1	EBPSO2	MEL- PSO	Lever- FLPSO
Glass	1.74	1.66	1.65	3.42	1.66	2.39	1.95	1.91	1.70	2.79	1.69	3.91	3.71	5.86
Wine	1.61	1.50	1.54	3.02	1.49	1.97	1.75	1.84	1.56	2.52	3.55	3.60	3.77	5.93
Heart	4.21	2.06	2.07	4.80	4.44	2.63	2.42	2.43	2.05	3.56	5.03	5.14	5.21	7.46
Zoo	2.98	1.15	1.14	2.61	2.64	1.54	1.40	1.58	1.19	1.86	2.57	2.54	2.85	5.52
Parkinsons	7.36	3.70	1.59	3.97	3.92	2.26	2.04	2.18	1.70	2.78	3.69	3.76	3.86	6.29
Dermatology	9.66	4.56	3.78	8.27	5.79	6.69	5.23	4.63	3.57	4.53	5.86	5.89	7.21	10.80
Ionosphere	9.95	4.51	3.48	8.83	6.56	6.57	5.18	3.76	2.94	4.20	5.61	5.86	5.17	10.37
LungCancer	2.82	0.92	1.00	2.47	2.01	1.60	1.21	1.82	1.04	1.34	1.80	1.94	1.81	7.00
Sonar	4.75	4.06	5.75	8.00	7.91	4.64	6.73	4.69	5.82	3.30	7.57	7.63	4.68	11.76
Musk1	11.29	4.02	4.45	10.83	9.28	6.70	4.91	7.01	4.20	5.94	12.45	13.25	8.77	23.06
Yale	23.86	3.92	3.42	5.47	8.56	11.33	6.30	14.44	5.37	6.11	7.36	8.04	6.47	77.03
SRBCT	22.77	7.22	7.69	9.48	15.54	17.72	12.99	27.06	8.82	10.67	12.35	13.17	10.08	149.28
warpAR10P	21.88	6.21	5.72	6.79	13.82	16.46	9.71	26.42	6.94	10.61	10.39	12.46	10.01	159.36
warpPIE10P	24.81	8.34	7.22	9.25	15.61	17.39	11.88	27.16	9.92	12.24	12.53	15.43	11.82	162.72
Lymphoma	57.23	13.28	8.75	14.86	12.55	50.32	26.49	83.55	18.76	24.57	15.26	16.64	22.62	160.18
TOX-171	120.32	17.10	15.86	18.94	33.68	47.20	29.70	73.77	22.44	96.05	26.52	31.80	21.85	466.28
9_Tumors	79.61	17.73	10.07	14.42	12.42	65.91	32.38	92.41	22.85	18.05	18.27	20.06	13.44	176.17
Leukemia	65.86	12.79	10.80	11.37	31.26	44.00	29.11	75.48	18.33	20.75	23.09	29.46	17.16	471.58
ALLAML	65.53	25.95	16.12	24.98	16.06	63.97	44.45	80.60	31.28	27.51	26.13	29.21	20.97	195.12
nci9	81.77	28.99	16.39	22.52	17.00	66.58	50.78	101.94	37.46	33.31	30.66	33.28	29.40	199.66
Adenocarcinoma	81.66	34.01	21.03	30.75	27.06	68.74	55.62	108.09	41.41	36.60	34.21	38.86	28.68	235.68
Prostate_Tumor	91.04	39.70	24.23	37.13	23.51	71.86	43.93	119.66	45.09	39.84	38.01	41.95	31.55	293.41

(Continued)

Table 9 (continued)

Name	BASO	ECSA	SSA	ISSA	mgFPA	TLPSO	COPSO	ASRPPSO	PLTVA CIW_PSO	SFE- PSO	EBPSO1	EBPSO2	MEL- PSO	Lever- FLPSO
11_Tumors	110.05	49.21	29.45	42.76	27.55	87.48	45.54	131.63	42.10	49.04	45.81	52.08	42.32	356.99
Lung_Cancer	120.71	57.82	36.03	56.51	32.59	95.00	52.77	138.13	43.38	53.73	49.82	52.71	44.48	293.17

4.3.2 Experimental Analysis

Based on the experimental results, it is evident that the proposed lever-FLPSO is an effective and reliable method. Lever-FLPSO significantly enhances search capabilities through lever-OBL, exploring as many high-quality feature subset combinations as possible to obtain the optimal solution. As demonstrated in the convergence graph for the “SRBCT” dataset (Fig. 5l), ISSA quickly falls into a local optimum, while lever-FLPSO continues to search and achieves the optimal result. Across 24 datasets, compared with 6 other PSO variants and 5 swarm intelligence algorithms, lever-FLPSO consistently selects the fewest number of features, ranking first overall. This success is attributed to the local selection and re-optimization (LSR) strategy, which significantly reduces the number of features while maintaining feature selection accuracy, thereby improving convergence speed. This effect is evident in the “Parkinsons”, “Dermatology”, “warpPIE10P”, and “Leukemia” datasets as shown in Figs. 4 and 5.

To mitigate the challenges associated with PSO, lever-FLPSO implements novel strategies that improve search efficiency, decrease the number of features, and expedite convergence speed, thereby effectively addressing several issues encountered by PSO in FS. Particularly, by applying fitness landscape (FL) in the selection of high-dimensional datasets, the search process has been optimized, thereby enhancing the interconnections among effective features in high-dimensional data. Furthermore, the integration of lever principle opposition-based learning (lever-OBL) facilitates a more effective aggregation of relevant information.

In the experiments, 80% of the feature subset from the optimal solution was selected as the new optimal solution (LSR) to reduce the number of features. The “80%” parameter was determined based on the algorithm’s performance across 24 datasets, where experiments were conducted using different percentages ranging from 10% to 100%, and the final average ranking was used to make the selection. As depicted in Fig. 10, the 80% setting demonstrated excellent performance in terms of both accuracy and the number of selected features. Consequently, 80% was chosen as the final parameter setting for the LSR strategy.

4.3.3 Sensitivity Analysis

This manuscript introduces three strategies (FL, lever-OBL, and LSR) and conducts experiments using 15 datasets to evaluate their effectiveness. The fundamental experimental parameters are utilized as outlined in Section 4.2, with adjustments only in the conditions of strategy application. As presented in Table 10, the integration of all three strategies in lever-FLPSO demonstrates complementary effects, achieving the highest ranking in both accuracy and selected feature count. It is evident from the table that the combination of lever-OBL and LSR surpasses the performance of the FL strategy when used independently, highlighting the valuable contributions of these strategies to the overall algorithm.

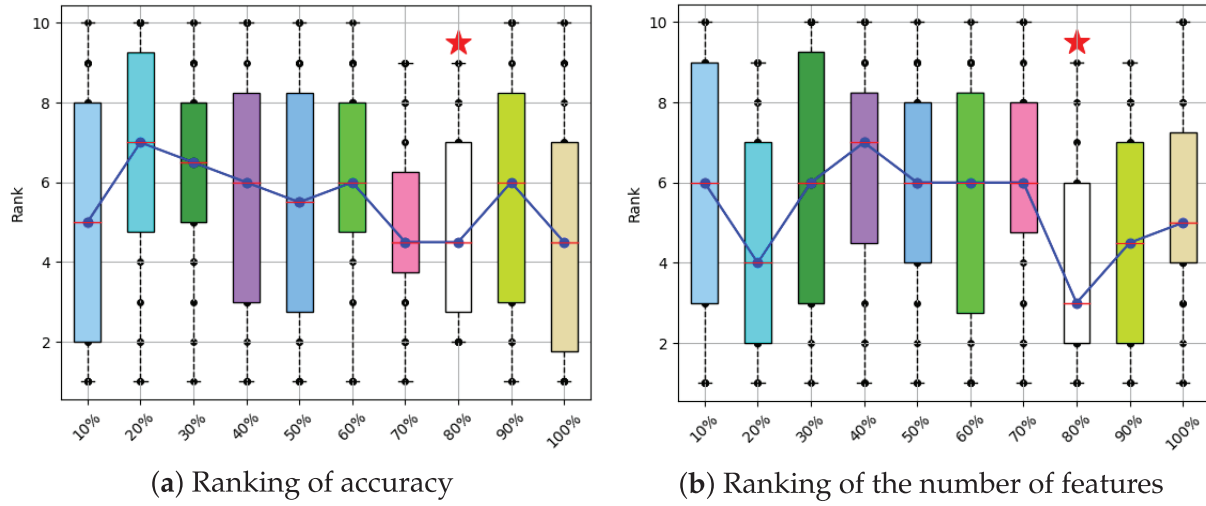


Figure 10: Analysis of LSR parameter setting

Table 10: The results of effectiveness analysis of strategies on the datasets

Name	Accuracy/Number of features			
	FL	FL+lever-OBL	FL+LSR	FL+lever-OBL+LSR
Glass	0.708/4.20	0.720/4.15	0.685/3.25	0.732/2.70
Wine	0.973/4.40	0.981/ 3.80	0.984 /4.00	0.981/3.90
Heart	0.614/4.35	0.600/4.25	0.621/4.25	0.641/3.20
Zoo	0.963/ 5.65	0.960/5.95	0.955/6.40	0.966 /6.20
Parkinsons	0.896/ 2.90	0.946 /3.35	0.902/3.55	0.942/3.30
Dermatology	0.989/14.50	0.973/13.35	0.996 / 11.55	0.995/12.60
Ionosphere	0.916/7.55	0.932 /9.10	0.917/7.50	0.929/ 7.30
LungCancer	0.972 /18.60	0.900/ 15.75	0.872/19.05	0.933/17.25
Sonar	0.902/22.60	0.898/22.30	0.900/23.35	0.947 / 22.10
Musk1	0.941 /72.80	0.887/67.85	0.936/70.75	0.919/ 63.25
Yale	0.624/390.95	0.759 /414.50	0.757/386.45	0.679/ 335.35
SRBCT	0.982/728.40	0.990/719.00	1.000 / 481.50	0.998/517.80
warpAR10P	0.441/ 571.55	0.645 /722.45	0.574/722.15	0.609/693.50
warpPIE10P	0.890/601.90	0.886/607.40	0.895/708.10	0.935 / 593.90
TOX-171	0.781/2351.95	0.882 /2628.90	0.712/2333.00	0.804/ 2285.90
Mean rank	3.00/2.93	2.60/2.73	2.60/2.73	1.80 / 1.53
Final rank	4/4	2/2	2/2	1/1

Note: The results of effectiveness analysis of strategies on the datasets, and the first-best is highlighted in bold.

As illustrated in Fig. 11, lever-FLPSO, which integrates the three strategies, exhibits superior and faster convergence in the “Glass”, “Heart”, “Sonar”, and “warpPIE10P” datasets. This improvement can be attributed to the FL strategy’s ability to balance the exploration and exploitation phases of the population, making the search process more conducive to convergence. The lever-OBL strategy further enhances the search process

by discovering more promising feature combinations, thereby circumventing local optima. As a concluding remark, the LSR strategy effectively selects the optimal feature subset, significantly reducing convergence time. No single strategy can solve all problems, as stated by the NFL theorem. The lever-FLPSO also faces challenges such as premature convergence, as shown in Fig. 11d. As demonstrated in Fig. 11c, although the incorporation of the LSR strategy into lever-FLPSO facilitates faster convergence relative to FL+lever-OBL, it may also lead to the omission of some potentially promising solutions due to the reduction in the number of features.

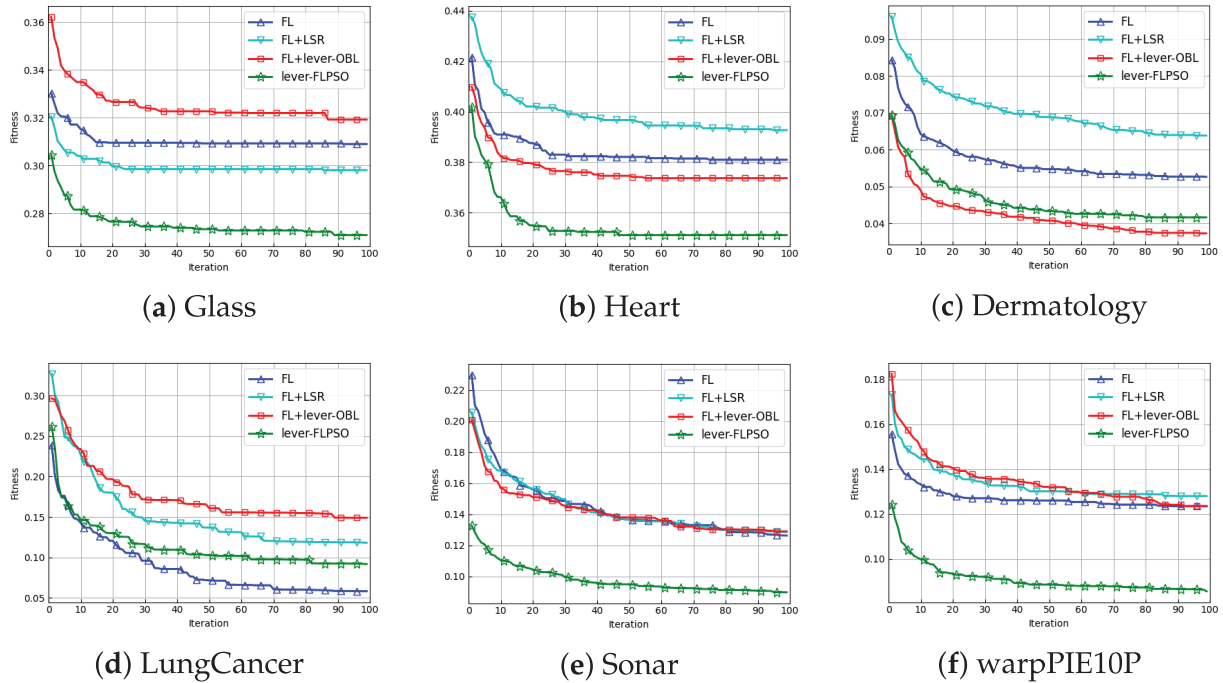


Figure 11: The convergence curves of 20 independent runs for lever-FLPSO and the ablation algorithms

5 Conclusion

In this article, a fitness landscape PSO algorithm based on leverage-based opposition learning was proposed for FS problems. The proposed lever-FLPSO adaptively adjusts different parameters based on the varying states of the population, balancing the exploration and exploitation phases to enhance the algorithm's search capability. The innovative lever-OBL strategy significantly increases the likelihood of discovering potentially optimal feature subsets. It also incorporates a LSR strategy to decrease the number of selected features, thereby enhancing convergence efficiency. By utilizing these sophisticated strategies, lever-FLPSO successfully navigates local optima and further augments its efficacy in FS challenges. The performance of the proposed lever-FLPSO was evaluated using 24 benchmark datasets and compared with 13 other FS algorithms. Overall, experimental results demonstrate that lever-FLPSO not only achieves high classification accuracy but also excels in minimizing the number of selected features.

Owing to the novel integration of lever-OBL and LSR within the algorithm, it is capable of more effectively identifying optimal feature combinations while simultaneously reducing the number of features in high-dimensional problems. The experimental data also demonstrate that the algorithm performs outstandingly in high-dimensional problems. Prospective avenues for this research can focus on further exploring its applications in high-dimensional problems. The concepts of the fitness landscape strategy and lever-OBL

can be extended to other heuristic algorithms to strengthen feature relationships in high-dimensional data and better integrate relevant information. However, employing the fitness landscape based on Euclidean distance increases the time complexity of the optimization process. Furthermore, the algorithm relies on the partitioned intervals of population states to guide the evolutionary direction, suggesting that optimizing these partitions remains a promising research direction. Therefore, further investigation into the formulation of the fitness landscape strategy is warranted.

Acknowledgement: Not applicable.

Funding Statement: This work was supported by National Natural Science Foundation of China (62106092); Natural Science Foundation of Fujian Province (2024J01822, 2024J01820, 2022J01916); Natural Science Foundation of Zhangzhou City (ZZ2024J28).

Author Contributions: Fei Yu: Conceptualization, Methodology, Data curation, Supervision, Writing—original draft, Writing—review & editing. Zhenya Diao: Conceptualization, Methodology, Data curation, Software, Writing—original draft, Writing—review & editing. Hongrun Wu: Supervision, Writing—review & editing. Yingpin Chen: Supervision, Writing—review & editing. Xuewen Xia: Supervision, Writing—review & editing. Yuanxiang Li: Supervision, Writing—review & editing. All authors reviewed the results and approved the final version of the manuscript.

Availability of Data and Materials: All data generated or analyzed in this study are published in this article.

Ethics Approval: Not applicable.

Conflicts of Interest: The authors declare no conflicts of interest to report regarding the present study.

References

1. Too J, Rahim Abdullah A. Binary atom search optimisation approaches for feature selection. *Connect Sci.* 2020;32(4):406–30. doi:10.1080/09540091.2020.1741515.
2. Dhal P, Azad C. A comprehensive survey on feature selection in the various fields of machine learning. *Appl Intell.* 2022;52(4):4543–81. doi:10.1007/s10489-021-02550-9.
3. Jiawei E, Zhang Y, Yang S, Wang H, Xia X, Xu X. GraphSAGE++: weighted multi-scale GNN for graph representation learning. *Neural Process Lett.* 2024;56(1):24. doi:10.1007/s11063-024-11496-1.
4. Cheng F, Cui J, Wang Q, Zhang L. A variable granularity search-based multiobjective feature selection algorithm for high-dimensional data classification. *IEEE Trans Evolut Comput.* 2023;27(2):266–80. doi:10.1109/TEVC.2022.3160458.
5. Chen K, Xue B, Zhang M, Zhou F. Evolutionary multitasking for feature selection in high-dimensional classification via particle swarm optimization. *IEEE Trans Evol Comput.* 2022;26(3):446–60. doi:10.1109/TEVC.2021.3100056.
6. Hashemi A, Dowlatabadi MB, Nezamabadi-pour H. MFS-MCDM: multi-label feature selection using multi-criteria decision making. *Knowl Based Syst.* 2020;206:106365. doi:10.1016/j.knosys.2020.106365.
7. Salesi S, Cosma G, Mavrovouniotis M. TAGA: tabu asexual genetic algorithm embedded in a filter/filter feature selection approach for high-dimensional data. *Inf Sci.* 2021;565(12):105–27. doi:10.1016/j.ins.2021.01.020.
8. Tran B, Xue B, Zhang M. Variable-length particle swarm optimization for feature selection on high-dimensional classification. *IEEE Trans Evol Comput.* 2019;23(3):473–87. doi:10.1109/TEVC.2018.2869405.
9. Nguyen BH, Xue B, Andreae P, Ishibuchi H, Zhang M. Multiple reference points-based decomposition for multiobjective feature selection in classification: static and dynamic mechanisms. *IEEE Trans Evolut Comput.* 2020;24(1):170–84. doi:10.1109/TEVC.2019.2913831.
10. Mistry K, Zhang L, Neoh SC, Lim CP, Fielding B. A micro-GA embedded PSO feature selection approach to intelligent facial emotion recognition. *IEEE Trans Cybern.* 2017;47(6):1496–509. doi:10.1109/TCYB.2016.2549639.

11. Selvaraj G, Subbiah J, Selvaraj M. Embedded binary PSO integrating classical methods for multilevel improved feature selection in liver and kidney disease diagnosis. *Int J Biomed Eng Technol.* 2019;31(2):105–36. doi:10.1504/IJBET.2019.102119.
12. Chen K, Xue B, Zhang M, Zhou F. An evolutionary multitasking-based feature selection method for high-dimensional classification. *IEEE Trans Cybern.* 2022;52(7):7172–86. doi:10.1109/TCYB.2020.3042243.
13. Too J, Abdullah AR. A new and fast rival genetic algorithm for feature selection. *J Supercomput.* 2021;77(3):2844–74. doi:10.1007/s11227-020-03378-9.
14. Tran B, Xue B, Zhang M. A new representation in PSO for discretization-based feature selection. *IEEE Trans Cybern.* 2018;48(6):1733–46. doi:10.1109/TCYB.2017.2714145.
15. Qiu C. A novel multi-swarm particle swarm optimization for feature selection. *Genet Program Evolvable Mach.* 2019;20(4):503–29. doi:10.1007/s10710-019-09358-0.
16. Zhang F, Mei Y, Nguyen S, Zhang M. Evolving scheduling heuristics via genetic programming with feature selection in dynamic flexible job-shop scheduling. *IEEE Trans Cybern.* 2021;51(4):1797–811. doi:10.1109/TCYB.2020.3024849.
17. Nag K, Pal NR. A multiobjective genetic programming-based ensemble for simultaneous feature selection and classification. *IEEE Trans on Cybern.* 2016;46(2):499–510. doi:10.1109/TCYB.2015.2404806.
18. Mirjalili S, Gandomi AH, Mirjalili SZ, Saremi S, Faris H, Mirjalili SM. Salp swarm algorithm: a bio-inspired optimizer for engineering design problems. *Adv Eng Softw.* 2017;114:163–91. doi:10.1016/j.advengsoft.2017.07.002.
19. Tizhoosh HR. Opposition-based learning: a new scheme for machine intelligence. In: *Proceedings of the International Conference on Computational Intelligence for Modelling, Control and Automation and International Conference on Intelligent Agents, Web Technologies and Internet Commerce (CIMCA-IAWTIC'06)*; 2005 Nov 28–30; Vienna, Austria. p. 695–701. doi:10.1109/CIMCA.2005.1631345.
20. Tubishat M, Idris N, Shuib L, Abushariah MAM, Mirjalili S. Improved salp swarm algorithm based on opposition based learning and novel local search algorithm for feature selection. *Expert Syst Appl.* 2020;145(13):113122. doi:10.1016/j.eswa.2019.113122.
21. Shambour MKY, Abusnaina AA, Alsalibi AI. Modified global flower pollination algorithm and its application for optimization problems. *Interdiscip Sci Comput Life Sci.* 2019;11(3):496–507. doi:10.1007/s12539-018-0295-2.
22. Mirjalili S, Lewis A. The whale optimization algorithm. *Adv Eng Softw.* 2016;95(12):51–67. doi:10.1016/j.advengsoft.2016.01.008.
23. Ouadfel S, Abd Elaziz M. Enhanced crow search algorithm for feature selection. *Expert Syst Appl.* 2020;159(1):113572. doi:10.1016/j.eswa.2020.113572.
24. Gu S, Cheng R, Jin Y. Feature selection for high-dimensional classification using a competitive swarm optimizer. *Soft Comput.* 2018;22(3):811–22. doi:10.1007/s00500-016-2385-6.
25. Khanesar MA, Teshnehlab M, Shoorehdeli MA. A novel binary particle swarm optimization. In: *Proceedings of the 2007 Mediterranean Conference on Control & Automation*; 2007 Jun 27–29; Athens, Greece. p. 1–6. doi:10.1109/MED.2007.4433821.
26. Chuang LY, Chang HW, Tu CJ, Yang CH. Improved binary PSO for feature selection using gene expression data. *Computat Biol Chem.* 2008;32(1):29–38. doi:10.1016/j.compbiolchem.2007.09.005.
27. Camacho-Villalón CL, Dorigo M, Stützle T. Exposing the grey wolf, moth-flame, whale, firefly, bat, and antlion algorithms: six misleading optimization techniques inspired by bestial metaphors. *Int Trans Operat Res.* 2023;30(6):2945–71. doi:10.1111/itor.13176.
28. Eberhart R, Kennedy J. A new optimizer using particle swarm theory. In: *Proceedings of the Sixth International Symposium on Micro Machine and Human Science*; 1995 Oct 4–6; Nagoya, Japan. p. 39–43. doi:10.1109/MHS.1995.494215.
29. Malan KM, Engelbrecht AP. A survey of techniques for characterising fitness landscapes and some possible ways forward. *Inf Sci.* 2013;241:148–63. doi:10.1016/j.ins.2013.04.015.
30. Harrison KR, Ombuki-Berman BM, Engelbrecht AP. A parameter-free particle swarm optimization algorithm using performance classifiers. *Inf Sci.* 2019;503(3):381–400. doi:10.1016/j.ins.2019.07.016.

31. Morgan R, Gallagher M. Length scale for characterising continuous optimization problems. In: Coello CAC, Cutello V, Deb K, Forrest S, Nicosia G, Pavone M, editors. Parallel problem solving from nature—PPSN XII. Berlin/Heidelberg, Germany: Springer; 2012. p. 407–16. doi:10.1007/978-3-642-32937-1_41.
32. Li W, Meng X, Huang Y. Fitness distance correlation and mixed search strategy for differential evolution. *Neurocomputing*. 2021;458(3):514–25. doi:10.1016/j.neucom.2019.12.141.
33. Yu F, Guan J, Wu H, Chen Y, Xia X. Lens imaging opposition-based learning for differential evolution with cauchy perturbation. *Appl Soft Comput*. 2024;152(1):111211. doi:10.1016/j.asoc.2023.111211.
34. Ma B, Xia Y. A tribe competition-based genetic algorithm for feature selection in pattern classification. *Appl Soft Comput*. 2017;58(9):328–38. doi:10.1016/j.asoc.2017.04.042.
35. Li T, Dong H, Sun J. Binary differential evolution based on individual entropy for feature subset optimization. *IEEE Access*. 2019;7:24109–21. doi:10.1109/ACCESS.2019.2900078.
36. Bi Y, Xue B, Zhang M. Using a small number of training instances in genetic programming for face image classification. *Inf Sci*. 2022;593(2):488–504. doi:10.1016/j.ins.2022.01.055.
37. Fang J, Wang Z, Liu W, Lauria S, Zeng N, Prieto C, et al. A new particle swarm optimization algorithm for outlier detection: industrial data clustering in wire arc additive manufacturing. *IEEE Trans Autom Sci Eng*. 2022;21(2):1244–57. doi:10.1109/TASE.2022.3230080.
38. Azar A, Khan ZI, Amin S, Fouad K. Hybrid global optimization algorithm for feature selection. *Comput Mater Contin*. 2023;74(1):2021–37. doi:10.32604/cmc.2023.032183.
39. Chen K, Zhou FY, Yuan XF. Hybrid particle swarm optimization with spiral-shaped mechanism for feature selection. *Expert Syst Appl*. 2019;128(1–2):140–56. doi:10.1016/j.eswa.2019.03.039.
40. Gupta S, Gupta S. Fitness and historical success information-assisted binary particle swarm optimization for feature selection. *Knowl Based Syst*. 2024;306(Mar):112699. doi:10.1016/j.knosys.2024.112699.
41. Beheshti Z. A fuzzy transfer function based on the behavior of meta-heuristic algorithm and its application for high-dimensional feature selection problems. *Knowl Based Syst*. 2024;284:111191. doi:10.1016/j.knosys.2023.111191.
42. Gao J, Wang Z, Jin T, Cheng J, Lei Z, Gao S. Information gain ratio-based subfeature grouping empowers particle swarm optimization for feature selection. *Knowl Based Syst*. 2024;286(2):111380. doi:10.1016/j.knosys.2024.111380.
43. Moradi P, Gholampour M. A hybrid particle swarm optimization for feature subset selection by integrating a novel local search strategy. *Appl Soft Comput*. 2016;43:117–30. doi:10.1016/j.asoc.2016.01.044.
44. Qiu C, Liu N. A novel three layer particle swarm optimization for feature selection. *J Intell Fuzzy Syst*. 2021;41(1):2469–83. doi:10.3233/JIFS-202647.
45. Too J, Sadiq AS, Mirjalili SM. A conditional opposition-based particle swarm optimisation for feature selection. *Connect Sci*. 2022;34(1):339–61. doi:10.1080/09540091.2021.2002266.
46. Tijjani S, Ab Wahab MN, Mohd Noor MH. An enhanced particle swarm optimization with position update for optimal feature selection. *Expert Syst Appl*. 2024;247(7):123337. doi:10.1016/j.eswa.2024.123337.
47. Ahadzadeh B, Abdar M, Safara F, Khosravi A, Menhaj MB, Suganthan PN. SFE: a simple, fast, and efficient feature selection algorithm for high-dimensional data. *IEEE Trans Evolut Comput*. 2023;27(6):1896–911. doi:10.1109/TEVC.2023.3238420.
48. Wang X, Shanguan H, Huang F, Wu S, Jia W. MEL: efficient multi-task evolutionary learning for high-dimensional feature selection. *IEEE Trans Knowl Data Eng*. 2024;36(8):4020–33. doi:10.1109/TKDE.2024.3366333.
49. Diao Z, Yu F, Guan J, Liu M. An enhanced particle swarm algorithm based on fitness landscape information. In: Proceedings of the 2024 IEEE Congress on Evolutionary Computation (CEC); 2024 Jun 30–Jul 5; Yokohama, Japan. p. 1–8. doi:10.1109/CEC60901.2024.10611936.
50. Zhan ZH, Zhang J, Li Y, Chung HSH. Adaptive particle swarm optimization. *IEEE Trans Syst Man Cybern Part B*. 2009;39(6):1362–81. doi:10.1109/TSMCB.2009.2015956.
51. Liu W, Wang Z, Zeng N, Yuan Y, Alsaadi FE, Liu X. A novel randomised particle swarm optimizer. *Int J Mach Learn Cybern*. 2021;12(2):529–40. doi:10.1007/s13042-020-01186-4.
52. Liu W, Wang Z, Yuan Y, Zeng N, Hone K, Liu X. A novel sigmoid-function-based adaptive weighted particle swarm optimizer. *IEEE Trans Cybern*. 2021;51(2):1085–93. doi:10.1109/TCYB.2019.2925015.

53. Yu F, Guan J, Wu H, Wang H, Ma B. Multi-population differential evolution approach for feature selection with mutual information ranking. *Expert Syst Appl.* 2025;260(9):125404. doi:10.1016/j.eswa.2024.125404.
54. Yang S, Wei B, Deng L, Jin X, Jiang M, Huang Y, et al. A leader-adaptive particle swarm optimization with dimensionality reduction strategy for feature selection. *Swarm Evol Comput.* 2024;91(9):101743. doi:10.1016/j.swevo.2024.101743.
55. Wang Y, Cai Z, Guo L, Li G, Yu Y, Gao S. A spherical evolution algorithm with two-stage search for global optimization and real-world problems. *Inf Sci.* 2024;665(2):120424. doi:10.1016/j.ins.2024.120424.
56. Markelle K, Rachel L, Kolby N. The UCI machine learning repository. 2025 [cited 2025 Aug 26]. Available from: <https://archive.ics.uci.edu>.
57. Li J, Cheng K, Wang S, Morstatter F, Trevino RP, Tang J, et al. Feature selection: a data perspective. *ACM Comput Surv.* 2017;50(6):45. doi:10.1145/3136625.