Reference Vector Guided Variables Selection for Expensive Large-scale Multiobjective Optimization

1st Jianqing Lin, 2nd Cheng He, 3rd Xueming Liu, 4th Linqiang Pan✉
1st, 3rd, 4th *School of Artificial Intelligence and Automation*
2nd *School of Electrical and Electronic Engineering*
Huazhong University of Science and Technology, Wuhan 430074, China
linjqcn@gmail.com, chenghe_seee@hust.edu.cn, xm_liu@hust.edu.cn, lqpan@mail.hust.edu.cn

***Abstract*—With the development of computer-aided engineering, various surrogate-assisted evolutionary algorithms (SAEAs) have been developed to solve the involved computationally expensive multiobjective optimization problems (EMOPs). With the increasing complexity of EMOPs, the number of decision variables has increased from tens to hundreds or even thousands. "Curse of Dimensionality" caused by large-scale decision space poses a great challenge to current SAEAs, which require massive function evaluations (FEs). We propose an SAEA with reference vector guided variables selection, namely RVSPSO, for solving large-scale EMOPs. Specifically, the reference vector guided variable selection strategy is proposed to select critical variables for optimization, which associates reference solutions for enhancing convergence and maintaining diversity. Meanwhile, a reference guided particle swarm optimization is proposed as the optimizer, where the velocities of particles are updated according to the directions pointing from the particles to the reference solution. This update strategy aims to accelerate the convergence rate in large-scale decision space. Moreover, radial basis function networks are used as surrogate models for efficient optimization. Experiments are conducted on large-scale EMOPs with up to 2000 decision variables. Experiment results show the proposed RVSPSO can effectively solve large-scale EMOPs to obtain well-converged and diverse solutions with limited FEs, compared with five state-of-the-art SAEAs.**

***Index Terms*—Large-scale multiobjective optimization, surrogate model, reference guided strategy, variables selection**

I. Introduction

Evolutionary algorithms (EAs) are naturally suitable for multiobjective and black-box optimization problems, attributing to their population-based and gradient-independent properties [1], [2]. However, EAs cannot be directly used to solve computationally expensive multiobjective optimization problems (EMOPs) in real-world applications since EAs require massive function evaluations (FEs) [3]–[5]. For instance, airfoil design often relies on computational fluid dynamics simulations, which can take hours to evaluate a single solution. Thus, computationally cheap surrogate models are used to predict the objective values or relationship of candidate individuals to replace some expensive function evaluations, which are widely embedded into EAs. Various machine learning models are used as surrogate models, such as Kriging model [6], radial basis function network (RBFN) [7], artificial neural network (ANN) [8], and support vector machine (SVM) [9]. EAs with the assistance of surrogate models are expected to be used to solve EMOPs with only limited FEs, called surrogate-assisted EAs (SAEAs) [10]. In recent decades, various SAEAs with different evolutionary optimization frameworks and surrogate models have been proposed. MOEA/D-EGO is an SAEA using a decomposition framework, which optimizes multiple single-objective optimization subproblems simultaneously, where Kriging is used as a surrogate model, and one Kriging is built for each objective function [11]. K-RVEA uses the reference-vector guided evolutionary algorithm (RVEA) [12] framework as the optimizer, which also employs Kriging as the surrogate model [13]. KTA2 uses different sampling strategies for managing two candidate archives and adopts an influential point-insensitive model to approximate the objective functions, resulting in significantly improving the prediction accuracy of surrogate models [14]. CPS-MOEA uses a classification-based preselection for multiobjective evolutionary algorithms, where the K-Nearest Neighbor (KNN) is used as the classifier to predict the dominance relationship of the candidate solutions [15]. Unlike CPS-MOEA, CSEA uses a feedforward neural network (FNN) as a classifier to predict the dominant relationship between a candidate solution and the reference solutions instead of one solution [16].

Despite the previously mentioned SAEAs have achieved competitive results in solving EMOPs with no more than 30 decision variables, they encounter challenges in solving high-dimensional EMOPs, attributed to the curse of dimensionality [17]. The main factors affecting the performance of existing SAEAs are the prediction accuracy of the surrogate model in high-dimensional decision space and the limited real FEs. Recently, some SAEAs have been proposed for solving challenging high-dimensional EMOPs. Integrated into an indicator-based MOEA with reference adaptation, EDN-ARMOEA uses a dropout neural network (DNN) to solve high-dimensional EMOPs with up to 100 decision variables [18]. Different from the "dropout" in EDN-ARMOEA, ADSAPSO uses an adaptive dropout mechanism to take advantage of the statistical differences between different solution sets in the decision space to guide the selection of some crucial decision variables, which uses RBFN as the surrogate models to approximate the objective functions [19]. MCEA/D, a recently proposed

This work was supported by the National Natural Science Foundation of China (No. 62172171 and No. U20A20306) and the Fundamental Research Funds for the Central Universities (HUST: 2023JYCXJJ011). (*Corresponding author: Linqiang Pan*)

classification-based SAEA, built multiple SVM classifiers in a decomposition-based MOEA to solve high-dimensional EMOPs, which has achieved excellent results while consuming limited real FEs [20].

The relevant settings of some representative SAEAs for expensive multi-/many-objective optimization are shown in Table I, where the "Usage" represents the usage of the surrogate models, i.e., approximation and classification, and the "Dimension" represents the numbers of the decision variables.

TABLE I: The Surrogate Model Settings and Problem Settings for Representative SAEAs.

Algorithm	Model	Usage	Dimension
MOEA/D-EGO [11]	Kriging	Approximation	{2, 6, 8}
K-RVEA [13]	Kriging	Approximation	10
KTA2 [14]	Kriging	Approximation	{9, 10, 11}
CPS-MOEA [15]	KNN	Classification	30
CSEA [16]	FNN	Classification	{10, 20, 30}
EDN-ARMOEA [18]	DNN	Approximation	{20, 40, 60, 100}
ADSAPSO [19]	RBFN	Approximation	{50, 100, 200}
MCEA/D [20]	SVM	Classification	{50, 100, 150}

Generally, SAEAs with Kriging models are used for solving low-dimensional EMOPs with no more than 30 decision variables, attributed to the low computational efficiency of Kriging models in high-dimensional decision space. In CPS-MOEA and CSEA, KNN and FNN are used as surrogate models for classification to solve EMOPs with up to 30 decision variables. In MCEA/D, SVM is used for classification to solve EMOPs with up to 150 decision variables. EDN-ARMOEA and ADSAPSO are used for solving EMOPs with up to 100 and 200 decision variables, respectively. As shown in the results in the literature, RBFN is more computationally efficient than DNN used in EDN-ARMOEA, which is more suitable for approximation in solving large-scale EMOPs.

Although these state-of-the-art SAEAs have been proven more effective in solving high-dimensional problems with limited real function evaluations, compared to those SAEAs designed for low-dimensional EMOPs, the convergence performance of existing SAEAs weakens as the number of decision variables in EMOPs increases, especially in large-scale EMOPs. Moreover, the challenge of population diversity in high-dimensional decision space remains, let alone in large-scale decision space. Taking the genes selection task [21] as an example, cells contain thousands of genes, which also makes classification algorithms time-consuming, posing challenges to many existing SAEAs.

To address the above challenges, we have proposed an SAEA with reference guided variables selection, named RVSPSO, for solving large-scale EMOPs. The main contributions of this work are summarized as follows:

1) A reference vector guided variables selection strategy is proposed for guiding the population to converge from different directions in the objective space, maintaining the diversity of solutions. Specifically, with the association of the reference solutions, only part of the decision variables that are more critical for optimization are selected by variable information comparison for further optimization.

2) A reference guided particle swarm optimization is proposed as the optimizer, where the velocities of particles are set to the vectors from the particles to the reference solution, accelerating the convergence of the solutions.

3) The effectiveness and efficiency of the RVSPSO are examined by comparing it with five state-of-the-art SAEAs on large-scale EMOPs with up to 2000 decision variables.

The rest of this paper is organized as follows. Section II introduces the RBFN and PSO. The details of RVSPSO are shown in Section III. Section IV gives the experimental setting and results. Section V summarizes the work.

II. Preliminaries

In this section, the surrogate model radial basis function network (RBFN) and the optimizer particle swarm optimization (PSO) are introduced.

A. Radial Basis Function Network

RBFN is one of the most commonly used surrogate models due to its simple structure and excellent approximation performance [22]. The structure of RBF consists of three layers: the input layer, the hidden layer, and the output layer, where the transformation function of the hidden unit is the radial basis function. The Gaussian kernel function is the most commonly used radial basis function, which can be expressed as

$$h_i(\mathbf{x}) = exp(-\frac{\|\mathbf{x}-\mathbf{c}_i\|}{2\sigma^2}), i = 1, 2, \ldots, n, \quad (1)$$

where $\mathbf{x}$ is an input vector, $\mathbf{c}_i$ is the center of the i-th basis function, σ is the smoothing parameter, and n represents the number of hidden units.

B. Particle Swarm Optimization

PSO is inspired by the regularity of the foraging behavior of flocks. In PSO, particles have two important properties: velocity and position. The velocity update for the i-th particle can be calculated as

$$\begin{aligned} v_{id}^{k+1} = {} & \omega v_{id}^{k} + c_1 r_1 (p_{id,\text{pbest}}^{k} - x_{id}^{k}) \\ & + c_2 r_2 (p_{d,\text{gbest}}^{k} - x_{id}^{k}), \end{aligned} \quad (2)$$

where k is the number of iterations, ω is the inertia weight, c_1 is the individual learning factor, c_2 is the population learning factor, r_1 and r_2 are random numbers between 0 and 1, v_{id}^k is the velocity of the i-th particle in the d-th dimension in the k-th iteration, x_{id}^k is the position of the i-th particle in the d-th dimension in the k-th iteration, $p_{id,\text{pbest}}^k$ is the historical optimal position of x_{id}^k, and $p_{d,\text{gbest}}^k$ is the historical optimal position of the d-th dimension.

The position of the d-th dimension of the i-th particle is updated as

$$x_{id}^{k+1} = x_{id}^{k} + v_{id}^{k+1}. \quad (3)$$

Due to its advantage of fast convergence, PSO is often used for solving optimization problems with high-dimensional decision variables [23], [24].

III. Proposed Algorithm

In this section, the main framework of the proposed RVSPSO is introduced, which is shown in Algorithm 1.

Algorithm 1: Framework of RVSPSO

Input : η (number of initial solutions), $MaxFEs$ (maximum number of FEs), ξ (maximum number of iterations), δ (control parameter), n (population size), d (number of decision variables), m (number of objectives), k (number of reference solutions).

Output : Evaluated solutions in A.

/* Initialization */

1 $A \leftarrow$ LHS$(\eta), FEs \leftarrow \eta$

2 $V \leftarrow$ Uniform_Point$(n, m), V' \leftarrow V$

/* Main Loop */

3 **while** $FEs \leq MaxFEs$ **do**

4 $\quad \beta \leftarrow 1 - \delta \times (FEs - \eta)/(MaxFEs - \eta)$

5 $\quad [R, \mathbf{c}] \leftarrow$ Reference_Selection(A, V', k)

6 $\quad A' \leftarrow \emptyset$

7 $\quad$ **for** $i = 1 : k$ **do**

8 $\quad\quad \mathbf{s} \leftarrow$ Variable_Selection$(A, \mathbf{c}, d, \beta)$ ◁ Alg. 2

9 $\quad\quad \mathcal{R} \leftarrow$ Train_RBFN$(A, \mathbf{s})$

10 $\quad\quad Q \leftarrow$ SARPSO$(A, \mathbf{s}, n, \mathcal{R}, \xi, \mathbf{r}_i)$ ◁ Alg. 3

11 $\quad\quad \mathbf{q} \leftarrow min_{\mathbf{q} \in Q}APD(\mathbf{r}_i, Q)$

12 $\quad\quad \mathbf{x}^* \leftarrow$ Variable_Recombination$(\mathbf{q}, \mathbf{r}_i, \mathbf{s})$

13 $\quad\quad A' \leftarrow A' \cup \mathbf{x}^*$

14 $\quad$ **end**

$\quad$ /* Update */

15 $\quad [A', V'] \leftarrow$ Evaluation&Updating(A', V)

16 $\quad A \leftarrow A \cup A', FEs \leftarrow FEs + k$

17 **end**

In the initialization, the archive A with η solutions is obtained by the Latin Hypercube Sampling (LHS) method (Line 1). A reference vector set V with approximately n uniformly distributed reference vectors is generated in m-dimensional objective space by $Uniform\_Point$ (the same as in MOEA/D [25]) and then copied to V' (Line 2).

In the main loop, β is set from 1 to $1-\delta$ and associated with the number of function evaluations (FEs), which controls the proportion of the number of selected variables to the number of total decision variables and the balance of convergence and diversity. A reference solution set $R = \{\mathbf{r}_1, \ldots, \mathbf{r}_k\}$ with k reference solutions is selected from A with the assistance of V' by $Reference\_Selection$ (Line 5). Without loss of generality, we use the reference solutions selection method in K-RVEA refer to [13]. Simultaneously, a label vector $\mathbf{c} = \{c_1, c_2, \ldots, c_{FEs}\}$ is obtained, where each element represents the label of a solution in A assigned to the corresponding reference solution.

For each reference solution $\mathbf{r}_i$, $\lfloor \beta \cdot d \rfloor$ decision variables $\mathbf{s}$ are selected from original d decision variables using the proposed reference vector guided variables selection method $Variable\_Selection$, where the selected variables are more critical for optimization with the association of the reference solutions. The details of the proposed reference vector guided variables selection will be shown in the subsequent subsection. m RBFNs are trained and put into a set $\mathcal{R}$, where the input is the solutions in A only containing selected decision variables in $\mathbf{s}$ and the objective values of solutions in A. The selected decision variables in $\mathbf{s}$ are optimized by the proposed reference guided particle swarm optimization with the assistance of the RBFNs $\mathcal{R}$, ($SARPSO$), obtaining a $\lfloor \beta \cdot d \rfloor$-dimensional offspring Q with selected decision variables in $\mathbf{s}$ and with the predicted objective values (Line 11), and more details will be illustrated in Section III-B. Next, the angle penalized distance (APD) between the reference solution $\mathbf{r}_i$ and individuals in Q is calculated, and the individual with the smallest APD value in Q in objective space is selected as the candidate individual $\mathbf{q}$ (Line 12).

The optimized individual $\mathbf{q}$ and the reference solution $\mathbf{r}_i$ are then recombined to form a candidate solution by the variable recombination strategy $Variable\_Recombination$ (Line 13). Specifically, the values of the decision variables at $\mathbf{s}$ in $\mathbf{r}_i$ are replaced with the values of the selected decision variables in $\mathbf{q}$, forming the candidate solution $\mathbf{x}^*$. More details of the variable recombination will be expanded in Section III-C.

When there are k candidate solutions in A', the real function evaluations are performed on the solutions in A'. Finally, the archive A and the reference vector sets V' are updated.

A. Reference Vector Guided Variable Selection

In the proposed RVSPSO, $\lfloor \beta \cdot d \rfloor$ decision variables are selected for further optimization, which can improve the convergence with the limited number of FEs. The details of the reference vector guided variables selection are shown in Algorithm 2.

Algorithm 2: Variable_Selection$(A, \mathbf{c}, d, \beta)$

Input : A (Archive), $\mathbf{c}$ (label vector of solutions), d (number of decision variables), β (proportion of the selected variables).

Output : the selected variables set $\mathbf{s}$.

1 $\hat{A} \leftarrow$ Solution_Assignment$(A, \mathbf{c})$

2 $[G, B] \leftarrow$ Solution_Division$(\hat{A})$

3 $[\bar{\mathbf{g}}, \bar{\mathbf{b}}] \leftarrow$ Mean_Variable(G, B)

4 $\bar{\mathbf{v}} \leftarrow$ Absolute_Difference$(\bar{\mathbf{g}}, \bar{\mathbf{b}})$

5 $\{\bar{v}'_1, \bar{v}'_2, \ldots, \bar{v}'_d\} \leftarrow$ Sort$(\bar{\mathbf{v}})$

6 $\mathbf{s} \leftarrow \{\bar{v}'_1, \bar{v}'_2, \ldots, \bar{v}'_{\lfloor \beta \cdot d \rfloor}\}$

Giving the label vector of solutions $\mathbf{c} = \{c_1, c_2, \ldots, c_{FEs}\}$, the solutions in the archive A with $c_j = i, j \in [1, FEs]$ are assigned to the current reference solution $\mathbf{r}_i$ and recorded as $\hat{A}$ (Line 1). In $\hat{A}$, the solutions in the first non-domination front are set to G, and the solutions in the last non-domination front are set to B. Fig. 1 shows an illustrative example of the selection of solutions (Line 2). Taking the solutions assigned to $\mathbf{r}_3$ as an example, the red solutions in the first non-domination front are set to G, and the green solutions in the last non-domination front are set to B.

Giving two solution sets G and B, the mean values of solutions' decision variables in G and B are calculated and recorded as $\bar{\mathbf{g}}$ and $\bar{\mathbf{b}}$ (Line 2). Then, execute a difference operation and an absolute value operation between $\bar{\mathbf{g}}$ and $\bar{\mathbf{b}}$ returning the absolute mean difference vector $\bar{\mathbf{v}}$, i.e., $\bar{\mathbf{v}} = |\bar{\mathbf{g}} - \bar{\mathbf{b}}|$. In the absolute mean difference vector $\bar{\mathbf{v}}$, the larger the

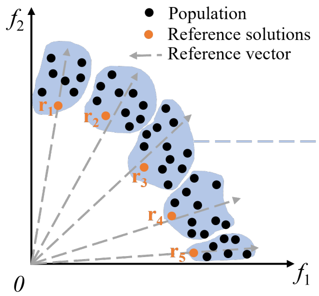


Fig. 1: An illustrative example of the selection of solutions, where the solutions in the same blue shaded area are assigned to the same reference solution, the red solutions in the first non-domination front are set to G, and the green solutions in the last non-domination front are set to B.

values, the corresponding decision variables are more critical to optimization with the association of the reference solutions. Therefore, the absolute mean difference values in $\bar{\mathbf{v}}$ are sorted in reverse order as $\{\bar{v}'_1, \bar{v}'_2, \ldots, \bar{v}'_d\}$ (Line 4). Specifically, the first $\lfloor \beta \cdot d \rfloor$ decision variables with larger absolute mean difference values $\{\bar{v}'_1, \bar{v}'_2, \ldots, \bar{v}'_{\lfloor \beta \cdot d \rfloor}\}$ are selected as $\mathbf{s}$ for further optimization. The remaining decision variables are not used for optimization.

B. SARPSO

The detailed descriptions of the surrogate-assisted reference guided PSO (SARPSO) are shown in Algorithm 3.

Algorithm 3: SARPSO($A, \mathbf{s}, n, \mathcal{R}, \xi, \mathbf{r}_i$)

Input : A (Archive), $\mathbf{s}$ (selected variable set), n (population size), $\mathcal{R}$ (RBFN set), ξ (maximum number of iterations), $\mathbf{r}_i$ (i-th reference solution).
Output : Offspring Q.

1 $\tilde{A} \leftarrow$ Archive_Transformation($A, \mathbf{s}$)
2 $P \leftarrow$ Environmental_Selection($\tilde{A}, n$)
3 [$\mathbf{p}^*$, $\mathbf{g}^*$] $\leftarrow$ Initialization(P)
4 **for** *iter* = $1 : \xi$ **do**
5     $Q \leftarrow$ RPSO($P, \mathbf{p}^*, \mathbf{g}^*, \mathbf{r}_i$)
6     $Q \leftarrow$ Surrogate_Evaluation($Q, \mathcal{R}$)
7     $[P, \mathbf{p}^*, \mathbf{g}^*] \leftarrow$ Update($P, Q, \mathbf{p}^*, \mathbf{g}^*$)
8 **end**

In SARPSO, archive A is transformed into sub-archive $\tilde{A}$, where the decision variables in $\tilde{A}$ are the selected decision variables in $\mathbf{s}$, and the objective values in $\tilde{A}$ are the same as in A (Line 1). Then, the initial population P with n solutions is selected by environmental selection, which refers to the one in NSGA-II [26]. $\mathbf{p}^*$ and $\mathbf{g}^*$ are the particle's best-known position and the swarm's best-known position in the population P, respectively. Afterwards, an offspring Q is generated using the proposed reference guided particle swarm optimization ($RPSO$) (Line 5). Different from the regular PSO, the velocities of particles in each iteration in $RPSO$ are set to the vectors pointing from the particles to the reference solution. Next, the individuals in Q are evaluated by the RBFNs $\mathcal{R}$ instead of the real FEs (Line 6). Finally, P, $\mathbf{p}^*$, and $\mathbf{g}^*$ are updated by using the updating strategy in MOPSO [27], until the maximum number of iterations ξ is reached.

C. Variable Recombination

After optimization by SARPSO, the solution with the smallest angle penalized distance (APD) between the reference solution $\mathbf{r}_i$ and offspring Q in objective space is selected as the best candidate individual $\mathbf{q}$. The values of the selected variables in the reference solution are replaced with the values in $\mathbf{q}$, obtaining the candidate solution $\mathbf{x}^*$ for re-evaluation. Specifically, an illustrative example of variable recombination is shown in Fig. 2.

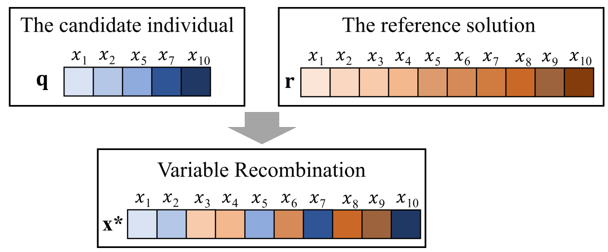


Fig. 2: An illustrative example of the variable recombination between the reference solution with 10 decision variables and the candidate individual with five selected variables.

The variable recombination of a 10-dimensional EMOP is shown in the illustrative example, where the number of selected variables is 5. Giving the best candidate individual $\mathbf{q}$ containing the selected variables $\mathbf{s} = \{x_1, x_2, x_5, x_7, x_{10}\}$, the candidate solution $\mathbf{x}^*$ is obtained by replacing the values of the selected variables of the reference solution $\mathbf{r}$ with the corresponding values in $\mathbf{q}$.

IV. Experimental Studies

To empirically investigate the performance of the proposed RVSPSO, it is compared with four state-of-the-art and representative SAEAs, namely CPS-MOEA [15], CSEA [16], EDN-ARMOEA [18], and ADSAPSO [19] on two test suites DTLZ and WFG with up to 2000 decision variables. In what follows, we briefly introduce the experimental settings of all the compared algorithms. Then, the selected test problems and performance indicators are described. Each algorithm is tested on each test instance over 20 independent runs to obtain the statistical results. Next, the Wilcoxon rank-sum test is used to compare the results achieved by the five compared algorithms at a significance level of 0.05. Specifically, symbols "+", "−", and "≈" represent that the compared algorithm is significantly better, significantly worse, and approximately equal to the proposed RVSPSO, respectively.

A. Experimental Settings

For fair comparisons, all the compared algorithms are implemented in the PlatEMO [28] on a PC with an AMD 5800x CPU processor and 32 GB RAM.[1]

[1]The code implementation of this work will be made available on GitHub (https://github.com/jqlincn/RVSPSO).

TABLE II: Statistics of IGD Results Achieved by CPS-MOEA, CSEA, EDN-ARMOEA, ADSAPSO, MCEA/D, and RVSPSO on 21 DTLZ test problems.

Problem	m	d	CPS-MOEA	CSEA	EDN-ARMOEA	ADSAPSO	MCEA/D	RVSPSO
DTLZ1	3	500	1.25e+4 (4.14e+2) -	1.41e+4 (3.96e+2) -	1.49e+4 (2.01e+2) -	1.45e+4 (5.81e+2) -	9.99e+3 (1.38e+3) -	6.05e+3 (7.63e+2)
		1000	2.56e+4 (4.76e+2) -	2.88e+4 (5.79e+2) -	3.03e+4 (4.40e+2) -	2.88e+4 (1.62e+3) -	1.69e+4 (2.68e+3) ≈	1.49e+4 (3.02e+3)
		2000	5.06e+4 (3.25e+3) -	5.90e+4 (8.29e+2) -	6.08e+4 (5.85e+2) -	5.95e+4 (3.29e+3) -	3.52e+4 (8.04e+3) -	2.39e+4 (3.46e+3)
DTLZ2	3	500	2.33e+1 (6.72e+0) -	3.54e+1 (8.81e-1) -	3.74e+1 (6.77e-1) -	3.08e+1 (3.32e+0) -	5.14e+0 (8.60e-1) -	4.17e+0 (5.24e-1)
		1000	4.23e+1 (2.61e+0) -	7.54e+1 (1.82e+0) -	7.74e+1 (8.93e-1) -	5.95e+1 (7.15e+0) -	2.06e+1 (4.96e+0) -	8.46e+0 (1.07e+0)
		2000	8.20e+1 (1.72e+0) -	1.56e+2 (1.84e+0) -	1.58e+2 (1.27e+0) -	1.21e+2 (1.49e+1) -	3.67e+1 (1.26e+1) -	1.42e+1 (1.62e+0)
DTLZ3	3	500	3.83e+4 (2.67e+3) -	4.67e+4 (9.17e+2) -	4.91e+4 (3.48e+2) -	3.56e+4 (6.68e+3) -	2.05e+4 (3.58e+3) -	1.51e+4 (2.54e+3)
		1000	7.63e+4 (4.67e+3) -	9.68e+4 (1.91e+3) -	1.01e+5 (6.79e+2) -	7.73e+4 (1.05e+4) -	4.09e+4 (5.65e+3) -	2.95e+4 (3.49e+3)
		2000	1.51e+5 (5.57e+3) -	2.01e+5 (2.08e+3) -	2.05e+5 (8.51e+2) -	1.55e+5 (1.81e+4) -	9.26e+4 (1.45e+4) -	5.92e+4 (1.33e+4)
DTLZ4	3	500	2.38e+1 (5.76e+0) -	3.39e+1 (7.87e-1) -	3.76e+1 (5.09e-1) -	3.16e+1 (1.65e+0) -	4.32e+0 (7.52e-1) ≈	4.66e+0 (5.28e-1)
		1000	4.45e+1 (8.17e+0) -	7.29e+1 (1.86e+0) -	7.74e+1 (1.25e+0) -	5.47e+1 (8.99e+0) -	1.63e+1 (3.85e+0) -	1.00e+1 (1.90e+0)
		2000	9.15e+1 (2.40e+1) -	1.51e+2 (1.97e+0) -	1.58e+2 (1.39e+0) -	1.31e+2 (7.92e+0) -	3.61e+1 (1.03e+1) -	2.02e+1 (3.84e+0)
DTLZ5	3	500	2.23e+1 (5.10e+0) -	3.56e+1 (1.41e+0) -	3.745e+1 (7.30e-1) -	2.94e+1 (2.33e+0) -	5.65e+0 (1.20e+0) -	4.42e+0 (7.21e-1)
		1000	4.63e+1 (1.35e+1) -	7.54e+1 (1.57e+0) -	7.73e+1 (7.07e-1) -	5.92e+1 (4.17e+0) -	1.62e+1 (4.71e+0) -	8.41e+0 (8.60e-1)
		2000	9.78e+1 (3.27e+1) -	1.56e+2 (1.90e+0) -	1.58e+2 (9.49e-1) -	1.17e+2 (1.48e+1) -	3.97e+1 (7.23e+0) -	1.73e+1 (3.81e+0)
DTLZ6	3	500	3.75e+2 (1.18e+1) -	4.41e+2 (1.81e+0) -	4.42e+2 (9.79e-1) -	3.82e+2 (1.57e+1) -	2.96e+2 (1.25e+1) -	2.68e+2 (1.35e+1)
		1000	7.65e+2 (1.79e+1) -	8.88e+2 (3.69e+0) -	8.91e+2 (2.10e+0) -	7.55e+2 (5.34e+1) -	5.59e+2 (2.88e+1) -	5.24e+2 (2.62e+1)
		2000	1.55e+3 (3.77e+1) -	1.78e+3 (3.14e+0) -	1.79e+3 (1.99e+0) -	1.54e+3 (6.04e+1) -	1.20e+3 (7.65e+1) -	1.11e+3 (9.39e+1)
DTLZ7	3	500	1.12e+1 (2.50e-1) ≈	9.98e+0 (3.33e-1) +	1.08e+1 (2.85e-1) +	1.01e+1 (4.42e-1) +	1.11e+1 (1.24e-1) ≈	1.11e+1 (3.08e-1)
		1000	1.13e+1 (2.16e-1) ≈	1.07e+1 (3.30e-1) +	1.11e+1 (1.39e-1) ≈	1.06e+1 (3.70e-1) +	1.12e+1 (2.43e-1) ≈	1.12e+1 (2.33e-1)
		2000	1.14e+1 (1.78e-1) ≈	1.11e+1 (2.79e-1) +	1.13e+1 (1.65e-1) ≈	1.10e+1 (1.42e-1) +	1.14e+1 (1.51e-1) ≈	1.13e+1 (1.89e-1)
"+/−/≈"			0/18/3	3/18/0	1/18/2	3/18/0	0/16/5	

1) Parameters Settings: The number of initial solutions for real FEs is set to 100 for all the compared algorithms, considering the large-scale decision space and computationally expensive evaluations [20]. In RVSPSO, the number of selected reference solutions is set to $k = 5$. The control parameter δ is set to 0.3. The recommended parameter settings in the literature are used for other compared algorithms.

2) Surrogate-assisted evolution: The maximum number of generations in the surrogate-assisted search is $\xi = 50$, which is set with reference to other SAEAs [29].

3) Termination Condition: The maximum number of real function evaluations is 300, considering the computationally expensive evaluations [20].

B. Test Problems and Performance Indicator

In empirical studies, we test seven DTLZ problems and nine WFG problems, where the number of decision variables is in $\{500, 1000, 2000\}$. DTLZ is one of the most commonly used test suites for multiobjective optimization algorithms, and WFG is a test suite usually designed for large-scale multi-objective optimization, where both the objective functions of DTLZ and WFG are assumed to be computationally expensive.

Regarding the performance assessment, the widely used Inverted Generational Distance (IGD) indicator is adopted, which assesses the quality of the obtained solution set in terms of convergence and distribution uniformity [30]. A smaller IGD value indicates a better algorithm performance.

C. General Performance

This section compares the proposed RVSPSO with five SAEAs, i.e., CPS-MOEA, CSEA, EDN-ARMOEA, ADSAPSO, and MCEA/D on DTLZ test problems and WFG test problems with 500, 1000, and 2000 decision variables. Statistical results on DTLZ and WFG problems are given in Table II and Table III, respectively. Notably, CPS-MOEA, CSEA, and MCEA/D are three popular classification-based methods, and EDN-ARMOEA and ADSAPSO are regression-based SAEAs, where EDN-ARMOEA, ADSAPSO, and MCEA/D are tailored for high-dimensional EMOPs.

As shown in Table II, RVSPSO has achieved 17 out of 21 best mean IGD results, followed by ADSAPSO (2 out of 21). ADSAPSO achieves the best results on DTLZ7 with 1000 and 2000 decision variables, CSEA achieves the best result in DTLZ7 with 500 decision variables, and MCEA/D achieves the best result on DTLZ4 with 500 decision variables. Neither CPS-MOEA nor EDN-ARMOEA achieves the best results on DTLZ test problems. However, RVSPSO can not achieve competitive results in DTLZ7.

As shown in Table III, RVSPSO has achieved 17 out of 27 best mean IGD results, followed by MCEA/D (5 out of 27) and CSEA (4 out of 27). CSEA achieves the best mean IGD results on WFG1, and MCEA/D achieves more competitive results on WFG5. EDN-ARMOEA achieves the best mean IGD results on WFG4 with 2000 decision variables. CPS-MOEA, EDN-ARMOEA, and ADSAPSO do not achieve the best results on any WFG test problems.

To visually show the capabilities of SAEAs in solving high-dimensional EMOPs, Fig. 3 and Fig. 4 plot the final solutions obtained by CPS-MOEA, CSEA, EDN-ARMOEA, ADSAPSO, MCEA/D, and RVSPSO on bi-objective DTLZ2 with 500 decision variables and bi-objective WFG3 with 1000 decision variables, where the red line denotes the PF. For the results given in Fig. 3, we can observe that RVSPSO has achieved the best-converged and diversity results. MCEA/D achieves the second-best result, but the diversity of its obtained solutions is severely lacking. ADSAPSO and CPS-MOEA have obtained several convergent solutions, but there is still a significant gap from the PF. CSEA has slight convergence trends in several directions, and EDN-ARMOEA has almost no significant convergence trend.

TABLE III: Statistics of IGD Results Achieved by CPS-MOEA, CSEA, EDN-ARMOEA, ADSAPSO, MCEA/D, and RVSPSO on 27 WFG test problems.

Problem	m	d	CPS-MOEA	CSEA	EDN-ARMOEA	ADSAPSO	MCEA/D	RVSPSO
WFG1	3	500	2.30e+0 (3.38e-2) -	1.79e+0 (1.33e-1) +	2.14e+0 (7.55e-2) ≈	2.23e+0 (1.98e-1) -	2.27e+0 (5.45e-2) -	2.08e+0 (1.51e-1)
		1000	2.31e+0 (4.63e-2) -	1.78e+0 (1.08e-1) +	2.18e+0 (7.77e-2) ≈	2.24e+0 (1.55e-1) ≈	2.26e+0 (9.61e-2) -	2.10e+0 (1.95e-1)
		2000	2.29e+0 (5.58e-2) ≈	1.75e+0 (8.42e-2) +	2.25e+0 (4.75e-2) ≈	2.18e+0 (2.22e-1) ≈	2.27e+0 (6.35e-2) ≈	2.19e+0 (1.67e-1)
WFG2	3	500	8.96e-1 (2.03e-2) -	8.16e-1 (1.40e-2) -	8.67e-1 (2.85e-2) -	9.03e-1 (5.74e-2) -	7.30e-1 (4.33e-2) ≈	7.47e-1 (6.79e-2)
		1000	9.02e-1 (2.11e-2) -	8.36e-1 (1.67e-2) -	8.86e-1 (3.02e-2) -	9.04e-1 (6.01e-2) -	8.22e-1 (5.13e-2) ≈	7.72e-1 (7.61e-2)
		2000	8.85e-1 (3.54e-2) -	8.52e-1 (2.67e-2) -	8.94e-1 (2.97e-2) -	8.91e-1 (5.24e-2) -	7.84e-1 (6.45e-2) ≈	7.72e-1 (4.74e-2)
WFG3	3	500	8.95e-1 (2.05e-2) -	8.72e-1 (7.07e-3) -	8.62e-1 (5.33e-3) -	8.75e-1 (7.78e-3) -	6.46e-1 (3.20e-2) -	5.18e-1 (2.51e-2)
		1000	8.91e-1 (1.86e-2) -	8.80e-1 (3.55e-3) -	8.72e-1 (1.86e-3) -	8.81e-1 (9.85e-3) -	7.10e-1 (5.88e-2) -	5.36e-1 (2.15e-2)
		2000	8.96e-1 (2.18e-2) -	8.83e-1 (3.85e-3) -	8.77e-1 (2.43e-3) -	8.89e-1 (4.49e-3) -	7.13e-1 (3.19e-2) -	5.25e-1 (2.13e-2)
WFG4	3	500	6.52e-1 (1.66e-2) ≈	5.90e-1 (2.70e-2) ≈	6.51e-1 (4.06e-2) ≈	6.93e-1 (5.26e-2) ≈	5.66e-1 (2.43e-2) +	6.36e-1 (6.40e-2)
		1000	6.50e-1 (2.24e-2) ≈	5.81e-1 (4.21e-3) ≈	6.45e-1 (3.36e-2) ≈	6.67e-1 (4.23e-2) ≈	5.90e-1 (1.77e-2) ≈	6.39e-1 (9.84e-2)
		2000	6.43e-1 (1.00e-2) -	6.10e-1 (6.89e-2) ≈	6.40e-1 (2.67e-2) ≈	6.96e-1 (8.84e-2) -	5.99e-1 (2.51e-2) ≈	6.06e-1 (5.12e-2)
WFG5	3	500	7.63e-1 (2.01e-2) ≈	7.89e-1 (8.89e-3) -	8.06e-1 (7.33e-3) -	7.72e-1 (4.15e-2) -	6.52e-1 (2.23e-2) +	7.34e-1 (4.08e-2)
		1000	7.57e-1 (1.22e-2) -	7.99e-1 (1.33e-2) -	8.14e-1 (6.70e-3) -	7.81e-1 (1.99e-2) -	6.13e-1 (2.26e-2) +	6.84e-1 (2.35e-2)
		2000	7.64e-1 (1.41e-2) -	8.01e-1 (8.87e-3) -	8.17e-1 (9.92e-3) -	7.87e-1 (1.33e-2) -	6.30e-1 (3.46e-2) +	7.11e-1 (2.56e-2)
WFG6	3	500	1.04e+0 (1.50e-2) -	9.86e-1 (7.99e-3) -	9.82e-1 (5.18e-3) -	9.82e-1 (3.19e-2) -	8.99e-1 (4.23e-2) -	8.02e-1 (3.87e-2)
		1000	1.05e+0 (1.73e-2) -	9.89e-1 (6.78e-3) -	9.90e-1 (4.53e-3) -	9.97e-1 (2.63e-2) -	9.62e-1 (4.23e-2) -	7.99e-1 (2.89e-2)
		2000	1.04e+0 (1.92e-2) -	9.96e-1 (4.60e-3) -	9.97e-1 (4.93e-3) -	1.00e+0 (2.78e-2) -	9.27e-1 (2.41e-2) -	8.29e-1 (2.82e-2)
WFG7	3	500	7.96e-1 (1.84e-2) -	7.58e-1 (1.04e-2) -	7.47e-1 (3.73e-3) -	7.44e-1 (1.94e-2) -	6.68e-1 (1.69e-2) -	6.29e-1 (2.32e-2)
		1000	7.98e-1 (1.51e-2) -	7.63e-1 (1.48e-2) -	7.50e-1 (5.79e-3) -	7.68e-1 (2.47e-2) -	7.04e-1 (1.31e-2) -	6.52e-1 (3.38e-2)
		2000	8.08e-1 (1.10e-2) -	7.65e-1 (6.08e-3) -	7.56e-1 (7.57e-3) -	7.69e-1 (1.76e-2) -	7.15e-1 (1.70e-2) -	6.45e-1 (2.73e-2)
WFG8	3	500	8.45e-1 (2.02e-2) -	7.93e-1 (1.65e-2) -	7.91e-1 (8.04e-3) -	8.15e-1 (1.79e-2) -	7.35e-1 (3.08e-2) -	6.91e-1 (2.04e-2)
		1000	8.41e-1 (1.45e-2) -	7.98e-1 (9.49e-3) -	7.86e-1 (1.40e-2) -	8.01e-1 (1.33e-2) -	7.52e-1 (1.93e-2) -	6.72e-1 (1.67e-2)
		2000	8.28e-1 (1.10e-2) -	7.86e-1 (1.28e-2) -	7.84e-1 (7.05e-3) -	7.99e-1 (3.04e-2) -	7.45e-1 (1.79e-2) -	6.71e-1 (2.79e-2)
WFG9	3	500	1.10e+0 (4.13e-2) -	1.02e+0 (2.22e-2) -	1.06e+0 (1.73e-2) -	9.96e-1 (3.96e-2) -	7.96e-1 (4.19e-2) -	6.87e-1 (4.82e-2)
		1000	1.10e+0 (3.34e-2) -	1.02e+0 (2.05e-2) -	1.05e+0 (1.63e-2) -	1.01e+0 (5.01e-2) -	9.45e-1 (4.05e-2) -	7.10e-1 (6.19e-2)
		2000	1.08e+0 (2.15e-2) -	1.01e+0 (1.33e-2) -	1.06e+0 (2.17e-2) -	1.03e+0 (3.41e-2) -	9.31e-1 (4.15e-2) -	6.82e-1 (4.95e-2)
"+/−/≈"			0/23/4	3/21/3	0/21/6	0/23/4	4/17/6	

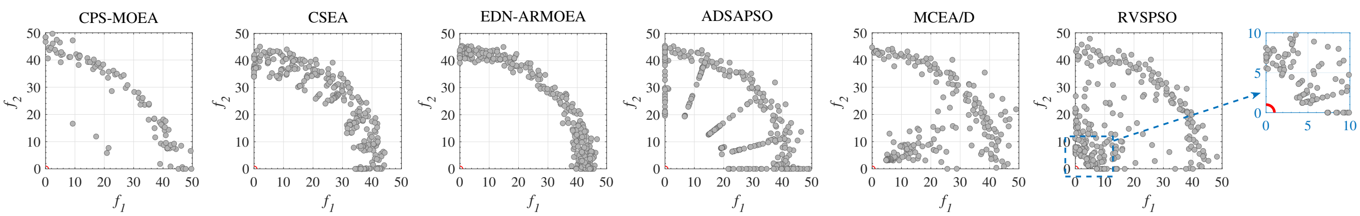


Fig. 3: Final solutions obtained by CPS-MOEA, CSEA, EDN-ARMOEA, ADSAPSO, MCEA/D, and RVSPSO on bi-objective DTLZ2 with 500 decision variables, where the red line denotes the PF.

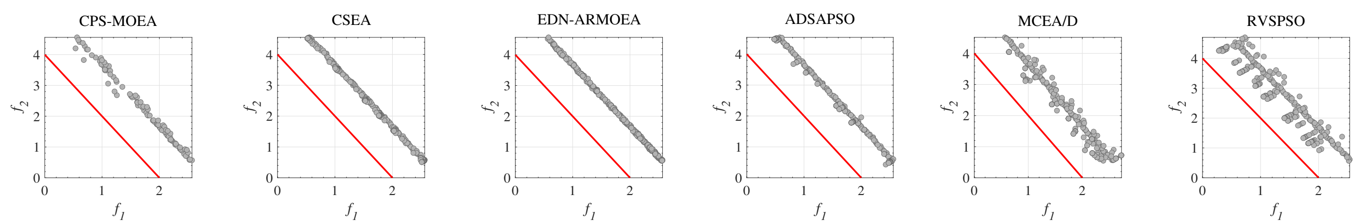


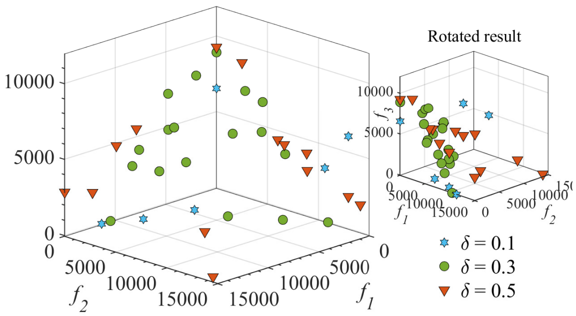
Fig. 4: Final solutions obtained by CPS-MOEA, CSEA, EDN-ARMOEA, ADSAPSO, MCEA/D, and RVSPSO on bi-objective WFG3 with 1000 decision variables, where the red line denotes the PF.

From the results given in Fig. 4, it can be observed that our proposed RVSPSO has obtained significantly convergent and diverse solutions, and MCEA/D has achieved the second-best result. CPS-MOEA, CSEA, EDN-ARMOEA, and ADSAPSO hardly ever obtain significantly converged solutions on WFG3 with 1000 decision variables.

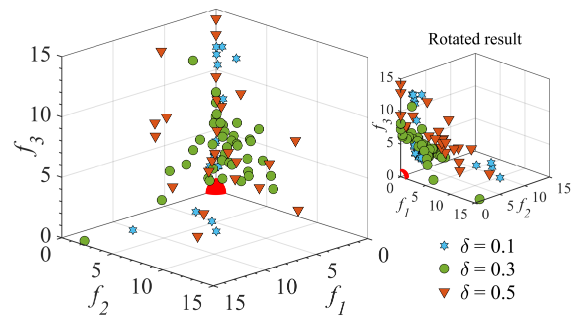
D. Parameter Sensitivity Analysis

To investigate the effectiveness of the RVSPSO with different δ values (the control parameter related to the number of selected variables), the RVSPSO with different δ values from $\{0.1, 0.3, 0.5\}$ are empirically compared. The non-dominated solutions obtained by RVSPSO with different δ values on 500-dimensional DTLZ1 and DTLZ2 are shown in Fig. 5.

We can observe that RVSPSO with $\delta = 0.1$ has achieved the most convergent solutions. However, the diversity of solutions obtained by RVSPSO with $\delta = 0.1$ is unsatisfactory, which only obtains a small number of the non-dominated solutions. The convergence of RVSPSO with $\delta = 0.5$ is the worst, since



(a) DTLZ1



(b) DTLZ2

Fig. 5: The non-dominated solutions obtained by RVSPSO with different δ values on DTLZ1 and DTLZ2 with 500 decision variables.

the non-dominated solutions obtained are the farthest from PF. RVSPSO with $\delta = 0.3$ is adopted as the default setting due to its superior balance between convergence and diversity.

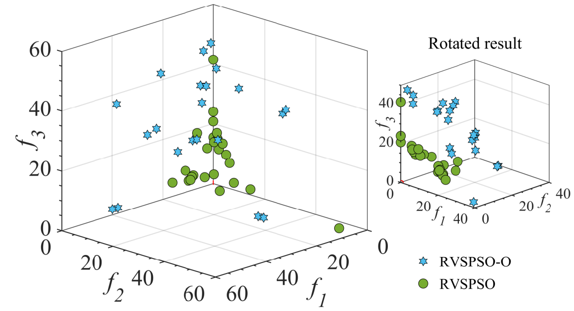


Fig. 6: The non-dominated solutions obtained by RVSPSO-O and RVSPSO on DTLZ5 with 2000 decision variables.

E. Effectiveness of the Reference Guided PSO

A variant of RVSPSO is compared to investigate the effectiveness of the proposed reference-guided PSO further. The designed variant, RVSPSO-O, uses regular PSO operations, the only difference from the proposed RVSPSO. The complete and rotated views of non-dominated solutions obtained by RVSPSO-O and RVSPSO on DTLZ2 with 2000 decision variables are shown in Fig. 6.

It can be observed that the non-dominated solutions obtained by RVSPSO have better convergence and diversity than the variant, RVSPSO-O, which shows the effectiveness of RVSPSO for solving large-scale EMOPs.

F. Convergence Speed

To investigate the convergence speed of the compared algorithms, the convergence profiles of CPS-MOEA, CSEA, EDN-ARMOEA, ADSAPSO, MCEA/D, and RVSPSO on seven DTLZ test problems with 2000 decision variables are displayed in Fig. 7.

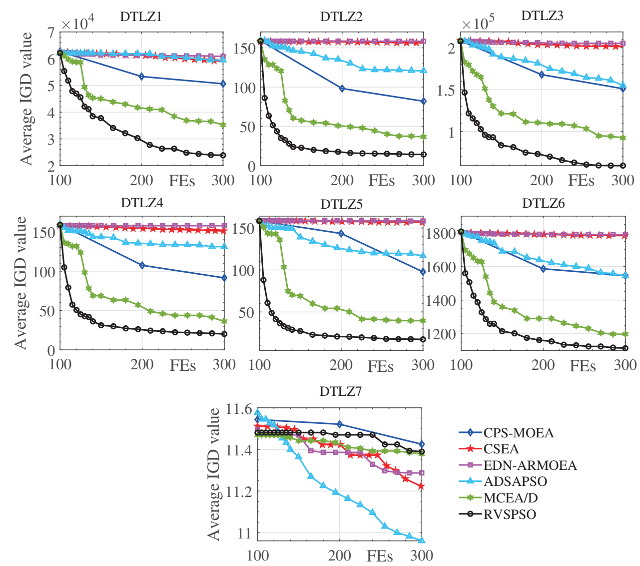


Fig. 7: The convergence profiles of CPS-MOEA, CSEA, EDN-ARMOEA, ADSAPSO, MCEA/D, and RVSPSO on DTLZ test problems with 2000 decision variables.

It can be observed that the proposed RVSPSO has shown superior convergence speed in DTLZ1-DTLZ6 with 2000 decision variables, followed by MCEA/D. However, RVSPSO does not converge significantly on DTLZ7, and ADSAPSO achieves the best convergence result on DTLZ7. Both CSEA and EDN-ARMOEA do not converge significantly in most DTLZ test problems.

G. Running Time

The running time of CPS-MOEA, CSEA, EDN-ARMOEA, ADSAPSO, MCEA/D, and RVSPSO on seven DTLZ test problems with 1000 decision variables are displayed in Fig. 8.

It can be observed that the running time of CPS-MOEA is smallest, and the one of EDN-ARMOEA is longest. The running time of RVSPSO is second longest, which may be caused by the search process assiseted with each reference point independently.

V. CONCLUSION

In this work, we propose an SAEA with reference vector guided variables selection, namely RVSPSO, for solving large-scale EMOPs with up to 2000 decision variables. A reference

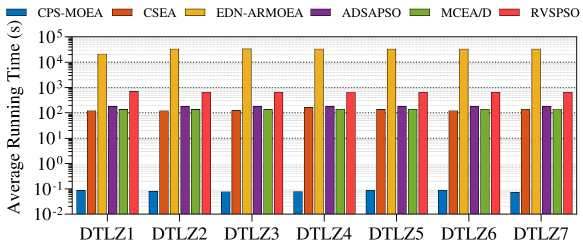


Fig. 8: The running time of CPS-MOEA, CSEA, EDN-ARMOEA, ADSAPSO, MCEA/D, and RVSPSO on DTLZ test problems with 1000 decision variables.

vector guided variables selection is proposed to select part of decision variables that are more critical for optimization with the association of the reference solutions, guiding the population to converge from different directions in the objective space. A reference guided particle swarm optimization is proposed to accelerate convergence, where the velocities of particles are set to the vectors from the particles to the reference solution. Moreover, radial basis function networks are used as surrogate models to approximate objective functions to assist the optimization. The proposed RVSPSO is compared with four state-of-the-art SAEAs to validate its effectiveness and efficiency in the experiments. The proposed RVSPSO can effectively solve large-scale EMOPs to obtain well-converged and diverse solutions.

While RVSPSO has demonstrated its superiority, significant insufficient convergence persists in some test problems. Future research should explore more effective variable selection strategies to address the challenges caused by the large-scale decision space. Besides, it is promising to extend RVSPSO to deal with real-world large-scale EMOPs.

References

[1] A. E. Eiben and J. Smith, "From evolutionary computation to the evolution of things," *Nature*, vol. 521, no. 7553, pp. 476–482, 2015.

[2] X. Wang, Z. Dong, L. Tang, and Q. Zhang, "Multiobjective multitask optimization-neighborhood as a bridge for knowledge transfer," *IEEE Trans. Evol. Comput.*, 2022.

[3] Y. Jin, "Surrogate-assisted evolutionary computation: Recent advances and future challenges," *Swarm Evol. Comput.*, vol. 1, no. 2, pp. 61–70, 2011.

[4] Q. Liu, R. Cheng, Y. Jin, M. Heiderich, and T. Rodemann, "Reference vector-assisted adaptive model management for surrogate-assisted many-objective optimization," *IEEE Trans. Syst. Man. Cybern. Syst.*, vol. 52, no. 12, pp. 7760–7773, 2022.

[5] L. Si, X. Zhang, Y. Tian, S. Yang, L. Zhang, and Y. Jin, "Linear subspace surrogate modeling for large-scale expensive single/multi-objective optimization," *IEEE Trans. Evol. Comput.*, 2023.

[6] J. Luo, A. Gupta, Y.-S. Ong, and Z. Wang, "Evolutionary optimization of expensive multiobjective problems with co-sub-pareto front gaussian process surrogates," *IEEE Trans. Cybern.*, vol. 49, no. 5, pp. 1708–1721, 2019.

[7] J. Park and I. W. Sandberg, "Universal approximation using radial-basis-function networks," *Neural Comput.*, vol. 3, no. 2, pp. 246–257, 1991.

[8] O. I. Abiodun, A. Jantan, A. E. Omolara, K. V. Dada, N. A. Mohamed, and H. Arshad, "State-of-the-art in artificial neural network applications: A survey," *Heliyon*, vol. 4, no. 11, p. e00938, 2018.

[9] S. Khandelwal, L. Garg, and D. Boolchandani, "Reliability-aware support vector machine-based high-level surrogate model for analog circuits," *IEEE Trans. Device Mater. Rel.*, vol. 15, no. 3, pp. 461–463, 2015.

[10] J. Knowles, "ParEGO: A hybrid algorithm with on-line landscape approximation for expensive multiobjective optimization problems," *IEEE Trans. Evol. Comput.*, vol. 10, no. 1, pp. 50–66, 2006.

[11] Q. Zhang, W. Liu, E. Tsang, and B. Virginas, "Expensive multiobjective optimization by MOEA/D with gaussian process model," *IEEE Trans. Evol. Comput.*, vol. 14, no. 3, pp. 456–474, 2010.

[12] R. Cheng, Y. Jin, M. Olhofer, and B. Sendhoff, "A reference vector guided evolutionary algorithm for many-objective optimization," *IEEE Trans. Evol. Comput.*, vol. 20, no. 5, pp. 773–791, 2016.

[13] T. Chugh, Y. Jin, K. Miettinen, J. Hakanen, and K. Sindhya, "A surrogate-assisted reference vector guided evolutionary algorithm for computationally expensive many-objective optimization," *IEEE Trans. Evol. Comput.*, vol. 22, no. 1, pp. 129–142, 2018.

[14] Z. Song, H. Wang, C. He, and Y. Jin, "A Kriging-assisted two-archive evolutionary algorithm for expensive many-objective optimization," *IEEE Trans. Evol. Comput.*, vol. 25, no. 6, pp. 1013–1027, 2021.

[15] J. Zhang, A. Zhou, and G. Zhang, "A classification and pareto domination based multiobjective evolutionary algorithm," in *IEEE Congr. Evol. Comput.* IEEE, 2015, pp. 2883–2890.

[16] L. Pan, C. He, Y. Tian, H. Wang, X. Zhang, and Y. Jin, "A classification-based surrogate-assisted evolutionary algorithm for expensive many-objective optimization," *IEEE Trans. Evol. Comput.*, vol. 23, no. 1, pp. 74–88, 2019.

[17] C. Sun, Y. Jin, R. Cheng, J. Ding, and J. Zeng, "Surrogate-assisted cooperative swarm optimization of high-dimensional expensive problems," *IEEE Trans. Evol. Comput.*, vol. 21, no. 4, pp. 644–660, 2017.

[18] D. Guo, X. Wang, K. Gao, Y. Jin, J. Ding, and T. Cai, "Evolutionary optimization of high-dimensional multiobjective and many-objective expensive problems assisted by a dropout neural network," *IEEE Trans. Syst. Man. Cybern. Syst.*, pp. 1–14, 2021.

[19] J. Lin, C. He, and R. Cheng, "Adaptive dropout for high-dimensional expensive multiobjective optimization," *Complex Intell. Syst.*, vol. 8, no. 1, pp. 271–285, 2022.

[20] T. Sonoda and M. Nakata, "Multiple classifiers-assisted evolutionary algorithm based on decomposition for high-dimensional multiobjective problems," *IEEE Trans. Evol. Comput.*, vol. 26, no. 6, pp. 1581–1595, 2022.

[21] X. Wang, T. Hu, and L. Tang, "A multiobjective evolutionary nonlinear ensemble learning with evolutionary feature selection for silicon prediction in blast furnace," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 33, no. 5, pp. 2080–2093, 2021.

[22] H.-M. Gutmann, "A radial basis function method for global optimization," *J. Glob. Optim.*, vol. 19, no. 3, pp. 201–227, 2001.

[23] X. Ji, Y. Zhang, D. Gong, and X. Sun, "Dual-surrogate-assisted cooperative particle swarm optimization for expensive multimodal problems," *IEEE Trans. Evol. Comput.*, vol. 25, no. 4, pp. 794–808, 2021.

[24] J. Tian, Y. Tan, J. Zeng, C. Sun, and Y. Jin, "Multiobjective infill criterion driven Gaussian process-assisted particle swarm optimization of high-dimensional expensive problems," *IEEE Trans. Evol. Comput.*, vol. 23, no. 3, pp. 459–472, 2018.

[25] Q. Zhang and H. Li, "MOEA/D: A multiobjective evolutionary algorithm based on decomposition," *IEEE Trans. Evol. Comput.*, vol. 11, no. 6, pp. 712–731, 2007.

[26] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," *IEEE Trans. Evol. Comput.*, vol. 6, no. 2, pp. 182–197, 2002.

[27] C. C. Coello and M. S. Lechuga, "Mopso: A proposal for multiple objective particle swarm optimization," in *IEEE Congr. Evol. Comput.*, vol. 2. IEEE, 2002, pp. 1051–1056.

[28] Y. Tian, R. Cheng, X. Zhang, and Y. Jin, "PlatEMO: A matlab platform for evolutionary multi-objective optimization [educational forum]," *IEEE Comput. Intell. Mag.*, vol. 12, no. 4, pp. 73–87, 2017.

[29] Y. Wang, J. Lin, J. Liu, G. Sun, and T. Pang, "Surrogate-assisted differential evolution with region division for expensive optimization problems with discontinuous responses," *IEEE Trans. Evol. Comput.*, vol. 26, no. 4, pp. 780–792, 2022.

[30] G. G. Yen and Z. He, "Performance metric ensemble for multiobjective evolutionary algorithms," *IEEE Trans. Evol. Comput.*, vol. 18, no. 1, pp. 131–144, 2014.