

# A Guidance Method for Robustness Surrogate Assisted Multi-objective Evolutionary Algorithms

Dinh Nguyen Duc<sup>1</sup>, Long Nguyen<sup>2</sup>, Hoai Nguyen Xuan<sup>3</sup>

<sup>1</sup> Academy of Military Science and Technology, Hanoi, Vietnam

<sup>2</sup> National Defense Academy, Hanoi, Vietnam

<sup>3</sup> Vietnam AI Academy, Hanoi, Vietnam

Correspondence: Dinh Nguyen Duc, nddinh76@gmail.com

Communication: received 23 February, revised 16 April, accepted 30 April

Digital Object Identifier: 10.32913/mic-ict-research.v2021.n1.948

**Abstract:** In the real world, multi-objective problems (MOPs) are relatively common in optimization in the areas of design, planning, decision support, etc. In fact, problems include two or many objectives, there is a class of problems called expensive problems that are problems with complex mathematical models, large computational costs, etc. They can not be solved by normal techniques, they are usually to be solved with techniques such as simulation, decomposing, problem transformation. In particular, using a surrogate model with Kriging, neural networks techniques in combination with an evolutionary algorithm is a subtle choice, with many positive results, being studied and applied in practice. However, the use of a surrogate model with Kriging, neural networks combining selection strategy, sampling... can reduce the robustness of the algorithms during the search. This paper analyzes the issues affecting the robustness of the multi-objective evolutionary algorithms (MOEAs) using surrogate models and suggests the use of a guidance technique to increase the robustness of the algorithm, through analysis, experiment and results are competitive and effective to improve the quality of MOEAs using a surrogate model to solve expensive problems.

**Keywords:** MOEA, robustness, surrogate, Kriging, FNN, CSEA, K-RVEA

## I. INTRODUCTION

In optimization studies including multi-objective optimization, the main focus is on finding global or Pareto-optimized solutions that represent the best objective values. In particular, algorithms using surrogate models, techniques are applied to reduce the number of calculations of the original objective function, increase the number of times using a surrogate objective function based on predicting progress of evolution. At that time, the user may not always be interested in finding the so-called global best solutions,

especially when these solutions are quite sensitive to the inevitable fluctuations in applying techniques in the use of surrogate models. In such cases, the problem of research is how to find powerful solutions that are less sensitive to disturbances from the characteristics of the technique used.

In this paper, we analyzed the effects of the mechanism using surrogate models in the multi-objective evolutionary algorithms, focusing on techniques using Kriging, neural network models, hypothesize and propose the use of automatic guidance techniques to create self-adaptive mechanisms for algorithms to improve the sustainability of the solution, convergence quality and diversity of the solution set, exploration and exploitation of the search process.

The remainder of the paper is organized as follows: A brief summary of surrogate assisted multi-objective evolutionary algorithms is given in Section II and the discussion of the robustness of MOEAs in Section III. Detail of our proposed guidance techniques are shown in Section IV. The experimental results are presented in Section V to examine the effectiveness and efficiency of proposed methods. Conclusion and future work are given in Section VI.

## II. SURROGATE ASSISTED MULTI-OBJECTIVE EVOLUTIONARY ALGORITHMS

### 1. Surrogate models

Surrogate models are used to approximate in simulation way to reduce the computational cost for expensive problems. The models are described as below:

If we call  $f(\vec{x})$  is an original fitness function of a MOP, then, we have  $f'(\vec{x})$  is a meta function, which is indicated as follows:

$$f'(\vec{x}) = f(\vec{x}) + e(\vec{x}) \quad (1)$$

function  $e(\vec{x})$  is the approximated error. In this case, the fitness function  $f(\vec{x})$  is not to be known, the values (input or output) are cared. Based on the responses of the simulator from a chosen dataset, a surrogate is constructed, then the model generates easy representations that describe the relations between preference information of input and output variables. There are some approaches for the surrogate models, which are divided into some kinds such as: the Radial Basis Function (RBF), the Polynomial Response Surface (PRS), the Support Vector Machine (SVM), neural network and the Kriging (KRG). The common concepts of using surrogate models are: instead of using known fitness functions always, to reduce the number of calculations, the evaluations are suggested to use surrogate functions, which are simpler. Therefore, without loss of generality, this paper investigates the use of the original and surrogate functions with Kriging model in the solving process of the MOEAs for expensive problems.

In [1], the authors proposed a method named "Kriging", which is a response surface method bases on spatial prediction techniques. This method minimizes the mean squared error to build the spatial and temporal correlation among the values of an attribute. In [2], the authors developed a parametric regression model which design and analysis of computer experiments, called DACE. The model is an extension of Kriging approach for at least three dimensions problems. The model is a combination of a known function  $f(x)$  and a Gaussian random process  $f'(x)$  that is assumed to have mean zero and co-variance, like as:

$$\begin{aligned} E(f'(\vec{x}^{(i)}), f'(\vec{x}^{(j)})) &= Cov(f'(\vec{x}^{(i)}), f'(\vec{x}^{(j)})) \\ &= \sigma^2 R(\vec{\theta}, \vec{x}^{(i)}, \vec{x}^{(j)}), \end{aligned} \quad (2)$$

here,  $\vec{x}^{(j)}$  is the correlation function with parameters  $\theta$ ,  $\sigma$  is the process variance of the response and  $R(\vec{\theta}, \vec{x}^{(i)}, \vec{x}^{(j)})$ .

With KRG model and popular MOEAs, there are some works, recently [3–8], for details:

In [4] the authors built a procedure which uses Kriging approximations and NSGA-II algorithm to optimize the aircraft design. The Kriging based genetic algorithm procedure gives shapes which are optimized for both the initial peak and perceived noise level of the signature along with minimum drag. The design problem concerns three objectives: the aircraft drag coefficient and the strength of the ground boom signature (using both the initial pressure rise and the perceived noise level) for each design point. In the study, Kriging approximation models are constructed from a large number of members of an initial population

using the results from repeated analyses carried out with BOOM-UA (a specified analysis tool). The approximation model is combined with the NSGA-II algorithm to minimize ground boom intensity while maintaining acceptable levels of aerodynamic cruise performance.

In [6], by providing a Design of Experiments (DOE) capability into the framework, the authors hybridized the desirable characteristics of evolutionary algorithms (EAs) and surrogate models such as RSM to obtain an efficient optimization system. Within this context, the DOE samples a number of design candidates at which the analysis code, the surrogate model is then constructed for the computationally expensive problem. Different sampling and DOE strategies can be used; Latin hypercube, RSM or DACE/Kriging.

In [7], in the problems with two objective functions, thermal resistance and pumping power have been selected to assess the performance of the micro-channel heat sink. The design variables related to the width of the micro-channel at the top and bottom, depth of the micro-channel, and width of fin, which contribute to objective functions, have been identified and a three-level full factorial design was selected to exploit the design space. The numerical solutions obtained at these design points were utilized to construct surrogate models Kriging and Radial Basis Neural Network. A hybrid multi-objective evolutionary algorithm coupled with surrogate models is applied to find out global Pareto-optimal solutions.

In [8], the author deeply discussed on surrogate MOEAs and indicated that: Kriging model appears to perform well in most situations, however it is much more computationally expensive than the rest. It is obvious that a careful consideration of RSM could lead to a situation where all objectives in a multi-objective problem are modeled using different methods, in order to maintain high quality and reduce optimization costs.

In [5] the KRG is used, the term Kriging points directly to the origin of these prediction methods dating back to the sixties, when the mining engineer Krige used Gaussian Random Field Models (GRFMs) to predict the concentration of ore in gold and uranium mines.

In [9], authors proposed such a method, MOEA/D-EGO, for dealing with expensive MOPs. The algorithm decomposes an MOP into a number of single-objective optimization sub-problems. At each iteration, a predictive distribution model is built for each individual objective in the MOP by using Fuzzy clustering and Gaussian stochastic process modeling. Then, a predictive model for the objective of each sub-problem can be induced. MOEA/D is used for maximizing the expected improvement metrics of all the sub-problems and several test points are then selected for

evaluation.

### III. ROBUSTNESS OF MULTI-OBJECTIVE EVOLUTIONARY ALGORITHMS

#### 1. General issues

The concept of robust algorithm was introduced in [10], there are two types of input for robustness system: control factors and noise factors, in what, the control factors can easily be controlled by decision maker and noise factors are hard to control of are beyond the control of a decision maker and they can be came from the environment. The methods used for robust algorithm can be broadly classified into two major groups: methods relying on gradient-based optimization for single optimization, and stochastic population-based methods for multi-objective optimization. Both use iteration-based approaches to improve solutions. In mathematical, two types of robust solutions are defined as below:

**Definition 1** (Type I): A solution  $\vec{s}^*$  is called robust solution of type I, if it is the global feasible Pareto-optimal solution to the following multi-objective minimization problem, in which it respect to a  $\delta$ -neighborhood  $\beta_\delta(\vec{s})$  of a solution  $\vec{s}$  in the global Pareto optimal set  $S$ :

$$\begin{aligned} & \text{minimize } (f_1^e(\vec{s}), f_2^e(\vec{s}), \dots, f_k^e(\vec{s})); \\ & \text{subject to } \vec{s} \in S \\ & \text{where } f_i^e(\vec{s}) \text{ is defined as follows:} \end{aligned} \quad (3)$$

$$f_i^e(\vec{s}) = \frac{1}{|\beta_\delta(\vec{s})|} \int_{\vec{y} \in \beta_\delta(\vec{s})} f_i(\vec{y}) dy$$

which  $|\beta_\delta(\vec{s})|$  denotes the volume of the neighborhood.

**Definition 2** (Type II): A solution  $\vec{s}^*$  is called a multi-objective robust solution of type II, if it is the global feasible Pareto optimal solution to following multi-objective minimization problem:

$$\begin{aligned} & \text{minimize } \vec{f}(\vec{s}) = (f_1(\vec{s}), f_2(\vec{s}), \dots, f_k(\vec{s})); \\ & \text{subject to } \vec{s} \in S \\ & \text{and } \frac{\|\vec{f}^p(\vec{s}) - \vec{f}(\vec{s})\|}{\|\vec{f}(\vec{s})\|} \leq \eta \end{aligned} \quad (4)$$

where  $\vec{f}^p(\vec{s})$  is the perturbed objective vector and it can be set, for example, as the worst function value in the neighborhood.  $\eta$  is predefined parameter, it is small real value.

In [11], in a general review, the authors analysed the depending on how robust the original global efficient front is respect to above definition, they can be the following four main scenarios:

- The complete original efficient front is robust: this case the original efficient front remains as an efficient

front with respect to the main effective objective functions. The entire set of original Pareto-optimal solutions is robust and is the goal of the optimization during the global efficient front constructed with the mean effective objectives are somewhat worse than that constructed with original ones.

- A part off the original efficient front is no more robust: this case, in such a problem, the task of a robust algorithm would be to identify only that part of the efficient front which is robust. In this case, the efficient front corresponding to the main effective objectives does not span over the entire original efficient region.
- The complete original global efficient front is non-robust; instead an original local efficient front is robust: this case, the global efficient front of the original problem is completely dominated by a local efficient front with respect to the mean effective objectives, thereby meaning that the original global Pareto-optimal solutions are not robust solutions and are sensitive to local perturbation. This case we need to find the robust efficient front for the problems.
- A part of the original global efficient front is robust together with a part of an original local efficient front.

There are many researchers worked on the analyzing robustness of multi-objective optimization. In [12], through an introduction of a measurement of the robustness of solutions in a MOEA, the authors also discussed on the effectiveness of combining the two approaches, different possibilities having been put forward and assessed by studying various multi-objective benchmark problems. In [13], the authors proposed a method to measure the multi-objective robustness of a design alternative using the sensitivity region concept and an approach using that measure to obtain robust Pareto solutions of multi-objective optimization problems. In [14], the author introduced an approach to compute solution paths for parameter-dependent multi-objective problems. The proposal combines techniques from numerical path following and from multi-objective optimization and allows to treat different kinds of problems. In [15], the authors focused on the (decision-maker's attitude to risk - DMAR) as a source of uncertainty and present two new types of robustness in MOP. In the first type, Dominance Robustness (DR), the robust Pareto solutions are those which, if perturbed, would have a high chance to move to another Pareto solution. In the second type, preference robustness (PR), the robust Pareto solutions are those that are close to each other in configuration space. The authors proposed methods to quantify these robustness concepts, modify existing evolutionary multi-objective optimization techniques to capture robustness against the DMAR, and present test problems to examine both DR and

PR.

## 2. Surrogate assisted multi-objective evolutionary algorithms issues

The method of using the surrogate model is a relatively effective and delicate choice in solving expensive problems. The techniques commonly used in the surrogate model are RBF, PRS, SVM, neural network and the Kriging,... combined with machine learning, deep learning, clustering, statistics, forecasting. However, determining the frequency of using the surrogate function to replace the original function, how to choose a reference solutions,... will create adaptive noises. These noises are treated as transformations of the surrogate function instead of the original function. This may cause a decrease in the optimal properties of the algorithm. Therefore, the use of a surrogate model can create a noisy environment for the search process of MOEA. Based on formula (1), the interference using the surrogate model can be defined as follows:

$$F_{noise} = f'(\vec{x}) = f(\vec{x}) + e(\vec{x}) \quad (5)$$

Here,  $e$  is also a noise, which is a noisy function. In the surrogate model, the difference between surrogate function values and the original fitness function ones is used to calculate the noise's standard deviation and noise's distribution.

The research issue we need to solve is keeping the robustness for the algorithm during the usage of the surrogate model. Focusing on the performance of the algorithm through basic measurements such as GD, IGD [16], we need to keep the minimized values all the time of searching. This means we need to feasible front to be closed and spread along the Pareto optimal front in every generation.

## IV. A GUIDANCE METHOD FOR SURROGATE ASSISTED ALGORITHMS

As hypothesized above, to reduce the noise of the environment, we need to maintain the population in the best of convergence and diversity during the search. Recently, many researchers discuss on the way of using the guidance method to drive the evolutionary process for the balance of exploitation and exploration of the process. It also maintains the convergence and diversity of the main population (obtained solutions) during the search continuously.

To build up the technique, we focus on two recent surrogate assisted multi-objective algorithms: K-RVEA [17] (using Kriging model) and CSEA [18] (using Feedforward Neural Networks - FNN). They are up to date surrogate assisted multi-objective algorithms, we will work on the algorithms with more investigation.

## 1. K-RVEA algorithm

Base on RVEA [19], a reference vector based MOEA for solving many-objective evolutionary problems (MaOPs), the authors proposed Kriging-assisted reference vector guided evolutionary algorithm (K-RVEA) [17]. In RVEA, search process is guided by a set of predefined reference vectors inspired from the direction vectors with the idea is to partition the objective space into a number small subspaces using a set of reference vectors. An elitism selection strategy is employed inside each subspace and using an Angle-Penalized Distance (APD) to measure the distance of the solutions to the ideal point and the closeness of the solutions to the reference vectors. An illustration of using reference vectors is shown in the Figure 1.

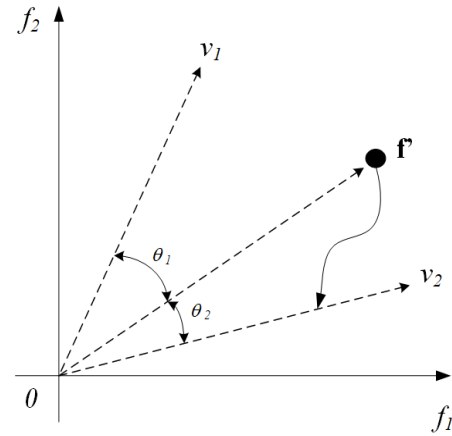


Figure 1. An example showing how to associate an individual with a reference vector. In this example,  $f'$  is a translated objective vector,  $v_1$  and  $v_2$  are two unit reference vectors.  $\theta_1$  and  $\theta_2$  are the angles between  $f'$  and  $v_1$ ,  $v_2$ , respectively. Since  $\theta_2 < \theta_1$ , the individual denoted by  $f'$  is associated with reference vector  $v_2$ .

In K-RVEA, based on RVEA, the authors introduced new algorithm which has three main phases: initialization, using and update the surrogate.

- 1) **Initialization:** The algorithm uses the Latin hypercube sampling to generate an initial population. Two archives  $A_1$  and  $A_2$  are filled with individuals which are evaluated with the original expensive functions. A Kriging model for each objective function is built by individuals which is given by the archive  $A_1$ .
- 2) **Using the surrogate:** This phase, the K-RVEA uses Kriging models instead of the original functions to calculate objective function values. Kriging models are used for fitness evaluations for a prefixed number of generations without updating them.
- 3) **Updating the surrogate:** After an evolutionary search using the Kriging models for a fixed number of generations, the Kriging models will be updated. The selection of individuals to be re-evaluated using

the original functions, which will also be used for updating the surrogates, is essential for the performance of surrogate-assisted evolutionary algorithms (SAEAs). The authors suggested to use information from the underlying evolutionary algorithm, RVEA, and uncertainty information from the Kriging models for selecting individuals to be re-evaluated and then for re-training the surrogate. The selected individuals are then re-evaluated with the original functions and these data samples are added to both archives  $A_1$  and  $A_2$ . The authors also use a reference vector based selection strategy to keep a fixed number of individuals in the archive  $A_1$ , it will eliminate extra individuals from  $A_1$ . By the way to assign reference vectors in various clusters see Figure 2, then randomly select one data point from each cluster and eliminate the rest of the data. This technique helps the algorithm to maintain a fixed number of diverse set of training data in  $A_1$ .

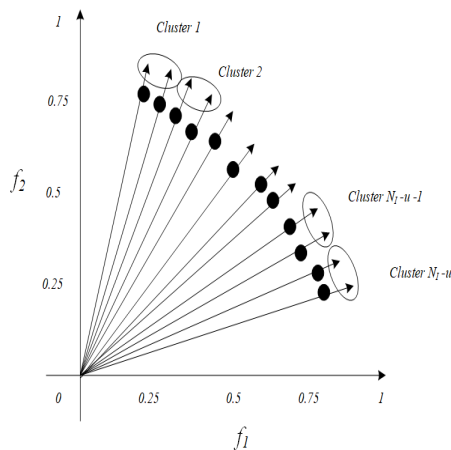


Figure 2. Clustering of active adaptive reference vectors  $V_\alpha$  into a predefined number of clusters  $u$ .

## 2. CSEA algorithm

The authors in [18] proposed a classification based surrogate-assisted multi-objective algorithm for expensive many-objective optimization called CSEA. A classification criterion is proposed in CSEA to divide solutions evaluated using the expensive objective functions into two different categories. Then a surrogate is employed for learning the classification criterion to predict candidate solutions in categories and use an selection strategy to select better converged solutions for the evaluations. In CSEA, the primary features are the classification criterion and the surrogate management strategy.

- 1) **Classification criterion:** The authors introduced a classification accuracy heavily depends on the classi-

fication criterion. Based on two types of solutions in MOPs: dominated and non-dominated, a set of reference solutions are chosen to construct the Pareto dominance boundary, which will divide all solutions into two different categories.

- 2) **The selection of reference solution:** To choose a set of reference solutions to form the classification boundary, a radial space division based selection strategy. In this selection strategy, all solutions evaluated using the expensive fitness function are first projected into a 2-dimensional radial space.

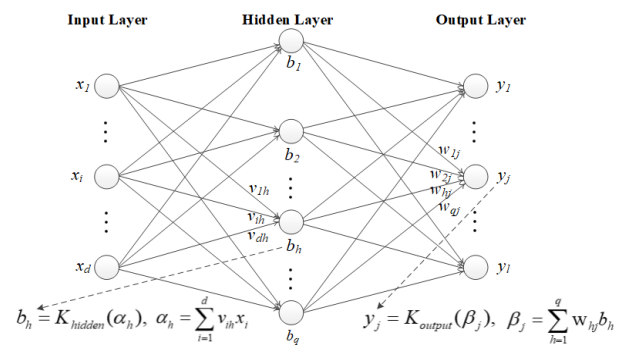


Figure 3. Illustration of a 3-layer FNN used in the CSEA algorithm.

To manage the surrogate model, the authors design the CSEA with four basic steps:

- **Initialization:** Initializing the parameters ( $K$ : number of reference solutions,  $gmax$ : number of solutions evaluated by surrogate model) and structure of the FNNs (which is one type of artificial neural networks in which connections between the neurons do not form a cycle). There are three main components of the FNN needed to be initialized: the network structure, the weights and the activation function.
- **Update:** Updating the weights of the FNN with the training data set. The authors used Levenberg-Marquardt back-propagation method to update the weights of the FNN.
- **Validation:** The authors used an error on test data as a measure to estimate the prediction uncertainty of the FNN. The errors are calculated by the Cross validation.
- **Surrogate assisted selection:** This step, some promising solutions are selected for the fitness evaluation. The selection is based on the predicted performance as well as the reliability. The selection used the errors on test data to estimate the reliability of the prediction of the FNN. A pair of errors (Test Error Rates of two categories) forms a point which is located in a region of the reliability configuration, the uncertainty of the FNN. The space presents the relationship between the uncertainty of the FNN and its test errors.

TABLE I  
GD VALUES OF K-RVEA AND M-K-RVEA ON DTLZS BENCHMARK SET

MOEAs	$w_{max}$	$E^1$									
		2000	4000	6000	8000	10000	12000	14000	16000	18000	20000
DTLZ1											
K-RVEA	10	50.7441	7.4836	5.6692	4.1103	3.2282	2.7263	2.5096	2.4711	2.4463	2.3877
	20	41.9546	10.0818	10.9039	3.7888	3.7041	3.7041	4.0672	4.0672	4.0097	4.0097
	30	39.0192	9.4198	7.2260	5.4170	3.9909	3.9062	3.9777	4.0149	3.9320	3.7725
	50	42.5498	10.4695	6.7609	5.9252	5.5349	4.6894	4.7492	4.3930	4.1622	4.0916
	70	43.5096	12.6410	8.0107	6.6715	7.1815	6.2105	5.7701	5.2643	5.0035	4.7697
M-K-RVEA	ADC	46.6079	4.7267	3.3725	3.3725	3.3725	3.3725	3.3725	3.3725	3.3725	3.3725
DTLZ2											
K-RVEA	10	0.1032	0.0023	0.0014	0.0011	0.0009	0.0008	0.0008	0.0007	0.0007	0.0006
	20	0.1151	0.0014	0.0009	0.0007	0.0006	0.0005	0.0005	0.0005	0.0004	0.0004
	30	0.0999	0.0015	0.0010	0.0008	0.0006	0.0005	0.0005	0.0004	0.0004	0.0004
	50	0.1076	0.0015	0.0009	0.0006	0.0005	0.0005	0.0004	0.0004	0.0004	0.0004
	70	0.1165	0.0014	0.0010	0.0007	0.0006	0.0005	0.0005	0.0004	0.0004	0.0004
M-K-RVEA	ADC	0.1227	0.0019	0.0011	0.0008	0.0007	0.0006	0.0006	0.0005	0.0005	0.0004
DTLZ3											
K-RVEA	10	183.1624	43.1944	36.1737	33.9130	30.8676	29.8677	28.4940	30.4472	29.3890	30.4415
	20	171.6720	65.1569	57.1465	47.8764	47.3865	42.2447	40.1520	38.8519	36.7741	36.5496
	30	220.4797	74.7820	56.8731	52.7536	52.7041	48.8366	44.5964	42.4959	41.5517	40.6848
	50	194.0912	85.5257	83.2413	70.2972	59.2193	50.7384	50.3928	49.8387	48.0394	44.5380
	70	151.5813	77.2795	68.4175	59.9071	51.6176	47.8923	49.5671	51.3997	48.6738	48.7996
M-K-RVEA	ADC	183.1624	69.5647	54.4809	48.0276	47.8869	43.2640	42.0306	40.3582	39.2478	42.3035
DTLZ4											
K-RVEA	10	0.1739	0.0148	0.0093	0.0063	0.0051	0.0043	0.0038	0.0034	0.0031	0.0029
	20	0.1613	0.0113	0.0072	0.0054	0.0044	0.0039	0.0035	0.0033	0.0031	0.0029
	30	0.1634	0.0108	0.0077	0.0060	0.0051	0.0044	0.0038	0.0035	0.0033	0.0031
	50	0.1841	0.0115	0.0074	0.0057	0.0048	0.0042	0.0038	0.0034	0.0031	0.0028
	70	0.1564	0.0129	0.0085	0.0073	0.0063	0.0055	0.0050	0.0047	0.0045	0.0042
M-K-RVEA	ADC	0.1564	0.0129	0.0085	0.0073	0.0063	0.0055	0.0050	0.0047	0.0045	0.0042
DTLZ5											
K-RVEA	10	0.1407	0.0068	0.0057	0.0054	0.0050	0.0046	0.0042	0.0042	0.0042	0.0041
	20	0.1571	0.0096	0.0069	0.0063	0.0056	0.0053	0.0052	0.0050	0.0049	0.0048
	30	0.1207	0.0111	0.0082	0.0068	0.0063	0.0058	0.0056	0.0054	0.0051	0.0051
	50	0.1584	0.0052	0.0046	0.0042	0.0041	0.0041	0.0040	0.0040	0.0039	0.0038
	70	0.1368	0.0101	0.0072	0.0065	0.0062	0.0056	0.0055	0.0052	0.0050	0.0049
M-K-RVEA	ADC	0.1286	0.0071	0.0055	0.0048	0.0045	0.0042	0.0039	0.0039	0.0037	0.0037
DTLZ6											
K-RVEA	10	0.9746	0.4591	0.4409	0.3822	0.3919	0.3913	0.3306	0.3261	0.3428	0.3140
	20	0.9513	0.4647	0.4897	0.4926	0.4734	0.4403	0.4151	0.3610	0.3625	0.3456
	30	0.9965	0.4503	0.3943	0.4227	0.3993	0.3845	0.3798	0.3657	0.3867	0.3529
	50	0.9480	0.5927	0.5124	0.4898	0.4478	0.4311	0.4444	0.4308	0.4108	0.3957
	70	0.9599	0.5421	0.4830	0.4446	0.4565	0.4716	0.4370	0.4339	0.4248	0.4184
M-K-RVEA	ADC	1.0474	0.4951	0.5234	0.5070	0.4695	0.4490	0.4279	0.4099	0.3823	0.3597
DTLZ7											
K-RVEA	10	2.0241	0.0026	0.0012	0.0008	0.0006	0.0005	0.0005	0.0004	0.0004	0.0003
	20	2.6543	0.0012	0.0007	0.0005	0.0004	0.0003	0.0003	0.0003	0.0002	0.0002
	30	2.6896	0.0009	0.0005	0.0004	0.0003	0.0003	0.0003	0.0002	0.0002	0.0002
	50	2.3290	0.0010	0.0007	0.0005	0.0004	0.0004	0.0003	0.0003	0.0002	2.0384
	70	2.0943	0.0007	0.0005	0.0005	0.0004	0.0003	0.0003	0.0003	0.0002	0.0002
M-K-RVEA	ADC	2.0241	0.0009	0.0007	0.0004	0.0004	0.0003	0.0003	0.0003	0.0002	0.0002
DTLZ8											
K-RVEA	10	0.0548	0.0322	0.0322	0.0322	0.0322	0.0322	0.0322	0.0322	0.0322	0.0322
	20	0.0587	0.0327	0.0327	0.0327	0.0327	0.0327	0.0327	0.0327	0.0327	0.0327
	30	0.0706	0.0358	0.0358	0.0358	0.0358	0.0358	0.0358	0.0358	0.0358	0.0358
	50	0.0728	0.0303	0.0303	0.0303	0.0303	0.0303	0.0303	0.0303	0.0303	0.0303
	70	0.0635	0.0357	0.0357	0.0357	0.0357	0.0357	0.0357	0.0357	0.0357	0.0357
M-K-RVEA	ADC	0.0635	0.0310	0.0310	0.0310	0.0310	0.0310	0.0310	0.0310	0.0310	0.0310
DTLZ9											
K-RVEA	10	3.4231	2.2998	1.5890	1.3484	1.4657	2.2714	1.4092	1.2228	1.1880	1.1553
	20	6.3959	2.5242	2.3244	2.1598	1.9384	1.9384	2.3200	1.5506	1.6016	2.1090
	30	4.2762	2.1689	1.9599	1.7168	1.6040	1.4022	1.5312	1.4316	1.3738	1.2709
	50	3.9907	1.5387	1.1807	0.9758	2.3940	2.1892	1.8939	1.8939	1.8430	1.7425
	70	4.5920	2.5457	1.9841	2.2016	2.0792	2.0511	1.9588	1.8728	1.9472	1.7881
M-K-RVEA	ADC	3.7733	2.5289	2.2851	2.1059	2.0260	2.0260	1.8907	1.8280	1.2761	1.1746

In the CSEA algorithm, there are two loops. The main loop represents the evolution using the original objective function. In the second loop, the solutions are selected through the classification of the surrogate model.

The experiments in [18] confirmed the author’s hypoth-

eses that surrogates classifying solutions into dominated and non-dominated solutions are promising for solving expensive many-objective optimization problems. CSEA is an up to date surrogate assisted multi-objective algorithm, so we build up our work on the algorithm with more investigation.

<sup>1</sup>Evaluations

TABLE II  
IGD VALUES OF K-RVEA AND M-K-RVEA ON DTLZS BENCHMARK SET

MOEAs	$w_{max}$	$E^1$	2000	4000	6000	8000	10000	12000	14000	16000	18000	20000
DTLZ1												
K-RVEA	10		37.5017	15.9551	15.9551	15.9551	8.6306	8.6306	8.6306	6.4437	6.4437	6.4437
	20		69.2926	12.2018	9.0805	6.0770	6.0770	6.0770	6.0770	6.0770	6.0770	6.0770
	30		73.5400	18.0276	18.0276	18.0276	18.0276	18.0276	18.0276	18.0276	18.0276	18.0276
	50		75.3314	21.8503	21.8503	21.8503	18.8011	18.8011	18.8011	18.8011	18.8011	18.8011
	70		56.6293	17.9351	17.9351	17.9351	15.7866	15.7866	11.9698	11.9698	11.9698	11.9698
M-K-RVEA	ADC		30.8156	9.9032	9.9032	9.9032	9.9032	9.9032	9.9032	9.9032	9.9032	9.9032
DTLZ2												
K-RVEA	10		0.4863	0.0523	0.0417	0.0372	0.0343	0.0326	0.0314	0.0302	0.0293	0.0284
	20		0.4862	0.0461	0.0368	0.0325	0.0297	0.0287	0.0281	0.0273	0.0265	0.0259
	30		0.4720	0.0467	0.0382	0.0326	0.0293	0.0283	0.0268	0.0260	0.0249	0.0244
	50		0.5233	0.0476	0.0387	0.0340	0.0306	0.0295	0.0285	0.0276	0.0272	0.0267
	70		0.4848	0.0502	0.0427	0.0372	0.0338	0.0324	0.0302	0.0292	0.0283	0.0272
M-K-RVEA	ADC		0.5617	0.0528	0.0398	0.0354	0.0335	0.0320	0.0305	0.0295	0.0287	0.0278
DTLZ3												
K-RVEA	10		500.9795	154.0806	154.0806	154.0806	154.0806	154.0806	154.0806	154.0806	154.0806	154.0806
	20		608.2876	195.9133	185.9003	185.9003	185.9003	185.9003	185.9003	185.9003	177.8553	177.8553
	30		421.7119	282.9219	229.2645	229.2645	229.2645	222.9678	132.3652	132.3652	132.3652	132.3652
	50		533.7593	229.4352	225.6308	225.6308	196.0629	196.0629	196.0629	196.0629	196.0629	195.8602
	70		672.9475	281.1927	274.5212	246.6838	246.6838	246.6838	183.5229	183.5229	183.5229	183.5229
M-K-RVEA	ADC		500.9795	235.2239	204.4000	204.4000	204.4000	182.6566	182.6566	182.6566	182.6566	182.6566
DTLZ4												
K-RVEA	10		0.9201	0.1337	0.0794	0.0595	0.0549	0.0495	0.0477	0.0458	0.0444	0.0434
	20		0.9733	0.0753	0.0606	0.0543	0.0503	0.0465	0.0447	0.0418	0.0408	0.0396
	30		0.8311	0.0745	0.0626	0.0568	0.0520	0.0466	0.0441	0.0421	0.0408	0.0399
	50		0.8441	0.0786	0.0637	0.0577	0.0541	0.0512	0.0474	0.0442	0.0420	0.0413
	70		0.8792	0.0833	0.0662	0.0629	0.0582	0.0552	0.0542	0.0524	0.0512	0.0504
M-K-RVEA	ADC		0.8792	0.0833	0.0662	0.0629	0.0582	0.0552	0.0542	0.0524	0.0512	0.0504
DTLZ5												
K-RVEA	10		0.3699	0.0395	0.0397	0.0384	0.0343	0.0330	0.0329	0.0327	0.0301	0.0301
	20		0.3739	0.0583	0.0525	0.0391	0.0371	0.0357	0.0355	0.0347	0.0345	0.0344
	30		0.4081	0.0722	0.0597	0.0496	0.0492	0.0478	0.0457	0.0449	0.0421	0.0421
	50		0.3483	0.0381	0.0337	0.0317	0.0307	0.0367	0.0364	0.0364	0.0364	0.0363
	70		0.3860	0.0494	0.0426	0.0412	0.0411	0.0394	0.0393	0.0388	0.0384	0.0376
M-K-RVEA	ADC		0.4247	0.0421	0.0371	0.0380	0.0373	0.0345	0.0338	0.0336	0.0333	0.0333
DTLZ6												
K-RVEA	10		8.2380	1.8545	0.8540	0.8110	0.8110	0.8110	0.8110	0.8110	0.7906	0.7724
	20		8.4386	2.2229	1.3579	1.3532	1.3532	1.2986	1.2986	1.2557	1.2557	1.2557
	30		8.2757	2.4312	2.4304	1.8914	1.7946	1.7637	1.6556	1.6556	1.6556	1.6437
	50		8.2283	2.7022	2.5015	2.4640	2.4640	2.4640	2.2781	2.2781	2.0575	2.0575
	70		8.4906	2.4451	2.4451	2.4451	2.1171	2.0384	2.0384	2.0384	2.0384	2.0384
M-K-RVEA	ADC		8.4051	2.9249	1.8047	1.6488	1.6488	1.6488	1.6488	1.6334	1.5731	1.5731
DTLZ7												
K-RVEA	10		8.1067	0.0611	0.0397	0.0332	0.0295	0.0276	0.0256	0.0243	0.0231	0.0222
	20		7.4668	0.0492	0.0406	0.0338	0.0317	0.0296	0.0280	0.0268	0.0256	0.0246
	30		8.6059	0.0514	0.0424	0.0380	0.0358	0.0341	0.0312	0.0303	0.0293	0.0282
	50		8.8847	0.0707	0.0599	0.0543	0.0469	0.0440	0.0420	0.0407	0.0398	0.0477
	70		8.2937	0.0725	0.0598	0.0559	0.0506	0.0487	0.0459	0.0442	0.0433	0.0424
M-K-RVEA	ADC		8.1067	0.0618	0.0546	0.0456	0.0433	0.0411	0.0359	0.0343	0.0320	0.0312
DTLZ8												
K-RVEA	10		0.2309	0.2050	0.2050	0.2050	0.2050	0.2050	0.2050	0.2050	0.2050	0.2050
	20		0.2590	0.1599	0.1599	0.1599	0.1599	0.1599	0.1599	0.1599	0.1599	0.1599
	30		0.2190	0.1827	0.1827	0.1827	0.1827	0.1827	0.1827	0.1827	0.1827	0.1827
	50		0.2516	0.2357	0.2357	0.2357	0.2357	0.2357	0.2357	0.2357	0.2357	0.2357
	70		0.2397	0.1935	0.1935	0.1935	0.1935	0.1935	0.1935	0.1935	0.1935	0.1935
M-K-RVEA	ADC		0.2397	0.1790	0.1790	0.1790	0.1790	0.1790	0.1790	0.1790	0.1790	0.1790
DTLZ9												
K-RVEA	10		11.3847	6.3491	6.1299	6.1299	5.6413	5.3711	4.9843	4.9843	4.9843	4.9843
	20		11.1618	6.3076	6.2910	6.2910	6.2910	6.2910	5.9713	5.9713	5.9713	5.9066
	30		11.1510	7.7752	7.7752	7.7563	7.7378	7.7212	7.7103	7.7103	7.7103	7.7099
	50		11.2579	7.9947	7.9363	7.9294	7.5453	7.5453	7.5453	7.5453	7.5453	7.5453
	70		11.1964	6.6274	6.3928	6.3837	6.3837	6.3837	6.3837	6.3837	6.3837	6.3830
M-K-RVEA	ADC		11.2529	6.5146	6.4943	6.4943	6.4943	6.4943	6.4943	6.4943	6.4697	6.4697

### 3. Noisy issues

In our work, we analysed the evolutionary process's behavior during the search with surrogate-assisted multi-objective evolutionary algorithms.

**K-RVEA:** For MOEA in general, according to the searching process, the quality of the achieved solution pop-

ulation varies according to the general rule, that is: in the first generations, the convergence quality of the algorithm is poor, the calculation quite diverse due to the fact that random generated solutions have not really evolved over generations. After that, the solution quality gradually got better, the diversity also gradually decreased and towards

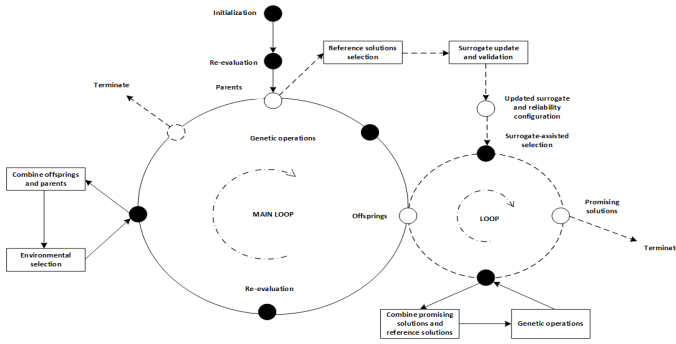


Figure 4. Demonstrate the main loop of evaluation using the original objective function and sub-loop of the selected solutions from the surrogate model.

the Pareto optimal front spread. The last generations of the process with a specified number of generations, the quality of the population converged was good because the process over many generations. An algorithm are assessed well if the diversity of the population is maintained when the solutions spread evenly over the Pareto optimization front. In many analyzes of the evolutionary trend of the process, in order to ensure a quality balance between convergence and diversity, it is necessary to guide the algorithm to keep a balance between exploration and exploitation cascade of evolution. Therefore, in the early stages, the ability to converge and the end stage need to enhance the diversity of the population. In K-RVEA, the authors use parameters to determine whether the model update is fixed, which can lack the algorithm’s adaptability, possibly making noises for environment which is against the robustness of the algorithm. We will test the K-RVEA algorithm on the class of expensive problems, with different parameters to evaluate the above comments.

We used  $w_{max}$ , the number of generations before updating the Kriging model with values: 10, 20, 30, 50 and 70 with DTLZs problems [20] (DTLZs from 1 to 9) in 20000 evaluations. We reported the results on GD and IGD, the popular metrics to measure the performance of the algorithm. The results are shown in Tables I, II. Some of the behaviors of the evolution process are reported in Figures 5 to 10.

**CSEA:** The strategy to select reference solutions is an important factor for the performance of the algorithm. The space division based on a boundary that is formed from the selected solutions. There is a key parameter for CSEA which is  $K$ : number of reference solutions that are selected. The role of  $K$  is discussed on the proposal. When value of  $K$  is small, the number of reference solutions of category II solutions is small, the algorithm has better convergence performance but is poor in terms of diversity. By contrast, that has better diversity but is poorly converged with larger

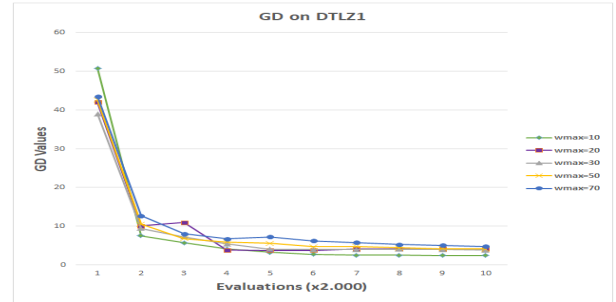


Figure 5. The behavior of K-RVEA on DTLZ1 (GD)

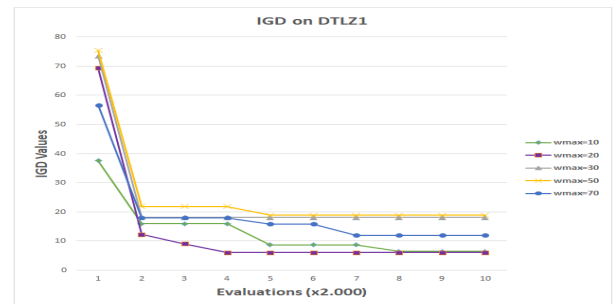


Figure 6. The behavior of K-RVEA on DTLZ1 (IGD)

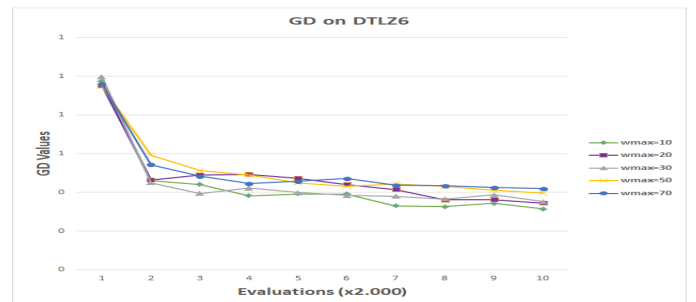


Figure 7. The behavior of K-RVEA on DTLZ6 (GD)

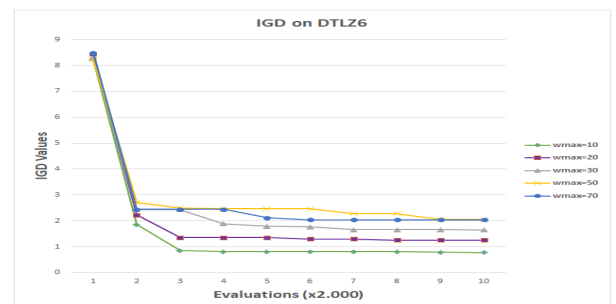


Figure 8. The behavior of K-RVEA on DTLZ6 (IGD)

value of  $K$ . The choice of the number of reference solutions provides a way of balancing the convergence and diversity of the selected category II solutions.

The authors also empirically investigate the impact of the number of reference solutions  $K$  on the performance



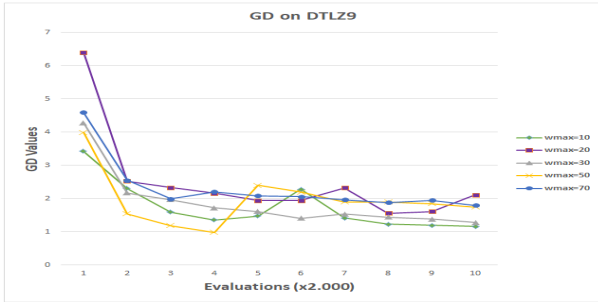


Figure 9. The behavior of K-RVEA on DTLZ9 (GD)

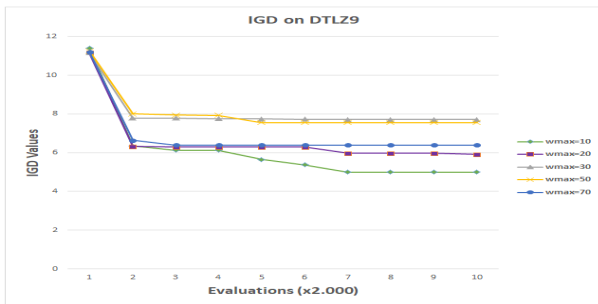


Figure 10. The behavior of K-RVEA on DTLZ9 (IGD)

of CSEA for solving problems with different numbers of objectives. CSEA with different settings of parameter  $K$  is tested on different benchmark sets. Through the results, the best choice of  $K$  is recorded to set for the algorithm. Then, the parameter is fixed to six for all the benchmark sets, it is known as a practical parameter value.

Similar to analysis of the evolutionary process's behavior in K-RVEA, the convergence quality in the early stages and the diversity of the population in the end stage need to be enhanced. In CSEA, the authors use  $K$  parameter to select reference solutions to form the classification boundary, a radial space division based on this selection strategy. Because the value of  $K$  is not adaptive, so it might lack the algorithm's adaptability, possibly making the noises for environment which is against the robustness of the algorithm.

#### 4. Guidance techniques

##### a) An adaptive control

Based on the analysis from the above experimental results with K-RVEA, for reducing the noise, the assumptions about appropriate parameter adjustment during evolution create an effective balance between convergence quality and diversity of the population. Specifically, adjusting the parameter on the number of generations before updating the model according to the assessed factors and assuming that the direct influence in SAEAs is: the processing time

of the evolution, the quality of the current population, and the computational characteristics of the problem.

Building the rate of using surrogate function and frequency of updating Kriging model flexibly, depending on the quality of the population in the current generation, to determine the quality of the solution, we use two defined values at a time  $t$ , with the test problem  $j$  as follows:

$$Q_t = \frac{n_{ND} * n_{gen}}{q_{size} * t_{size}} C_j \quad (6)$$

In which,  $n_{ND}$  is the number of non-dominated solutions in the main population,  $q_{size}$  is the size of the population, current generation  $n_{gen}$ ,  $t_{size}$  is the number of generations.  $C_j$  is the complexity of the problem, this is a parameter dependent on the complexity of the test function, it depends on the size of decision spaces, number of objectives and number of calculations for objective functions, we suggest to choose practical values  $C_j \in [0.8, 1]$ , the default value is 1. The constant 60 is used, which is the upper range of  $w_{max}$ , it helps the control to get the maximized value, 50. Based on the  $Q_t$  value, we will calculate the value of  $w_{max} \in R$  (the rate of using surrogate function and the frequency of updating the Kriging model) in the range [10, 50], a normalize function is defined to adjust valid  $w_{max}$  values.

$$w_{max} = \text{normalize}(Q_t * 60) \quad (7)$$

The modified version M-K-RVEA is described in Algorithm 1.

##### b) An adaptive number of reference solutions

Analyzing from the experimental results with CSEA, we hypothesize that, in order to reduce the noise by keeping balance the evolution between exploration and mining as well as to maintain the convergence and diversity of the population, the selection of a reference solution requires adaptability. It depends on factors that we can identify with the following influences:

- The process of evolution: evolution takes place in the process from the initial state of creation, the central stage and the final stage, the quality of the main population (number of non-dominated solutions) at each stage are different. Therefore, the use of reference solutions should be adjusted according to the quality of the current population in terms of convergence, diversity. In other words, at each time, it is necessary to select the number of different reference solutions to create a balance between exploration and exploitation, convergence and diversity.
- The sample problems are different, the quality of the population in each stage has different development, so at each specific time, based on the quality of the population to choose the appropriate number of reference

TABLE III  
DTLZS PROBLEMS FOR EXPERIMENTS

Name	Specific characteristics
DTLZ1	The Pareto front is linear (Hyper-plane). There are $(11^k - 1)$ local optimal fronts where $k$ is a user-specified parameter
DTLZ2	For $M > 3$ , the Pareto optimal solutions lie inside the first quadrant of the unit sphere in a three-objective plot with $f_M$ as one of the axes
DTLZ3	There are $(3^k - 1)$ local fronts that are parallel to the global Pareto front where $k$ is a user-specified parameter
DTLZ4	The Pareto optimal solutions are non-uniformly distributed along the Pareto front
DTLZ5	The front is a curve and the Pareto optimal solutions are non-uniformly distributed along the Pareto front
DTLZ6	A more difficult version of the DTLZ5 problem: the non-linear distance function $g$ makes it harder to convergence against the pareto optimal curve
DTLZ7	The Pareto front is formed by $2^{M-1}$ disjoint regions in the objective space
DTLZ8	The Pareto front is a combination of a straight line and a hyper-plane. The straight line is the intersection of the first $(M - 1)$ constraints with $f_1 = f_2 = \dots = f_{M-1}$ and the hyper-plane is represented by another constraint $g_M$
DTLZ9	The Pareto front is a curve with $f_1 = f_2 = \dots = f_{M-1}$ . The solution density gets thinner towards the Pareto front

solutions. It also makes the algorithm highly adaptive in maintaining the balanced convergence quality and diversity of the population.

- The difficulty of the problems is different, making predictions about the difficulty of the problem, deciding on the reference solution from users is also an important factor to guide the evolution flexibly in maintaining a balance between exploration and mining. This can both help the problem create leaps that can overcome local points, and the combination of decision makers and evolution.

Based on the analysis from the above experimental results, the assumptions about appropriate parameter adjustment during evolution is to create an effective balance between convergence quality and diversity of the population. Specifically, adjusting the parameter on the number of generations before updating the model according to the assessed factors and assuming that the direct influence in algorithms using a surrogate function is: the process time of the process evolution, the quality of the current population and the computational characteristics of the problem.

Designing a dynamic selection strategy to select reference solutions to form the boundary for the classification, we take into account the influence of the quality of the solution population in the current generation. Similar to the adaptive control technique for K-RVEA, to determine the quality of the solution, we calculate  $Q_t$  value by using equation 6. Based on the  $Q_t$  value, we will calculate the value of  $K \in R$  (the number of reference solutions will be selected to form the boundary) in the range [3, 9], a normalize function is defined to adjust valid  $K$  values. We use constant 6 as a generated point to control the  $K$  in the above range.

$$K = \text{normalize}(Q_t * 6) \quad (8)$$

The modified version M-CSEA is described in Algorithm 2.

## V. EXPERIMENTAL STUDIES

### 1. Testing problems and parameter settings

In the experiments, we used the origin K-RVEA and the modified version M-K-RVEA with the usage of the above adaptive  $w_{max}$ , and the origin CSEA and the modified version called M-CSEA with the usage of the above dynamic  $K$ . We also used some selected expensive problems in DTLZs [20] benchmark set. In the benchmark set, we have multiple kinds of expensive problems, which can be described in Table III.

In our experiments, the parameters are: the distribution index of crossover  $n_c = 20$  and the distribution index of mutation  $n_m = 20$ , the crossover probability  $pc = 1.0$  and the mutation probability  $p_m = 1/d$ , where  $d$  is the number of decision variables. Each case study, we run 30 independent times for analysing experimental results.

### 2. Performance metrics

Performance metrics are usually used to compare algorithms in order to form an understanding of which algorithm is better and in what aspects. However, it is hard to define a concise definition of algorithmic performance. In general, when doing comparisons, a number of criteria are employed [16]. Since each metric has both advantage and disadvantage, we will look at all two popular criteria: the generational distance (GD) and the inverse generational distance (IGD).

The GD measure is defined as the average distance from a set of solutions, denoted  $P$ , found by evolution to the global Pareto optimal set (POS) [21]. The first-norm equation is defined as:

$$GD = \frac{\sum_{i=1}^n d_i}{n} \quad (9)$$

where  $d_i$  is the Euclidean distance (in objective space) from solution  $i$  to the nearest solution in the POS, and  $n$  is the size of  $P$ . This measure is considered for convergence

**Algorithm 1: M-K-RVEA**

- 
- 1 Input:**
- 2 The maximum number of expensive function evaluations:  $FE_{max}$ ; the number of individuals to be re-evaluated and to be used for updating Kriging models is  $u$ ; the number of generations before updating Kriging models:  $w_{max}$ .
- 3 Output:**
- 4 Set of non-dominated solutions which are evaluated from  $A_2$  set.
- 5 */\* Initialization \*/*
- 1) Initialize a new population of size  $N_I$  with Latin hypercube sampling, set the number of function evaluations  $FE = 0$ , initialize the counters: the generation counter for using Kriging models  $w = 1$ , the number of updates  $t_u = 0$ . Set archives  $A_1$  and  $A_2$  to be empty,  $|A_1| = |A_2| = \emptyset$
  - 2) Evaluate the initial population with the original objective functions and add them to  $A_1$  and  $A_2$ , update  $FE = FE + N_I$ ,  $|A_1| = |A_1| + N_I$  and  $|A_2| = |A_2| + N_I$
  - 3) Using archive  $A_1$  to train a Kriging model for each objective function.
  - 4) **while**  $FE < FE_{max}$   
*/\* Using Kriging models \*/*
  - 5) Calculate  $Q_t$  by equation 6 then re-calculate  $w_{max}$  by equation 7
  - 6) **while**  $w \leq w_{max}$
  - 7) Run RVEA steps
    - Generate offspring population
    - Combine parent and offspring populations
    - Select parents from the population combined for the next generation.
    - Update reference vectors
    - Update  $w = w + 1$
  - 8) **end while**  
*/\* Updating Kriging models \*/*
  - 9) Select  $u$  individuals according to either the amount of uncertainty or the value of angle penalized distance (APD) [17] and re-evaluate them with the original objective functions and update  $FE = FE + u$ . Add these individuals to two archives  $A_1$ ,  $A_2$  and update  $|A_1| = |A_1| + u$  and  $|A_2| = |A_2| + u$
  - 10) Remove  $|A_1| - N_I$  individuals from  $A_1$  with specified concept of managing training data archive [17], update  $w = 1$  and  $t_u = t_u + 1$  and go to step 3.
  - 11) **end while**
- 

**Algorithm 2: M-CSEA**

- 
- 1 Input:**
- 2 Population size:  $N$ ; the number of reference solutions:  $K$ ; the number of hidden neurons:  $H$ ; the maximum number of fitness evaluations (FEs):  $t_{max}$ ; the maximum number of surrogate predictions before being updated:  $g_{max}$
- 3 Output:**
- 4 Set of non-dominated solutions from population  $P$ .
- 5 */\* Initialization \*/*
- 1) Initialize  $P$  with  $11d - 1$  solutions using Latin hypercube sampling method.
  - 2) Set  $t = 11d - 1$  with number of decision variables  $d$ .
  - 3) Initialize the neural network with  $H$  hidden neurons with  $m$  is the number of objectives
  - 4) Copy individuals from  $P$  to the archive  $Arc$ .
  - 5) **while**  $t \leq t_{max}$
  - 6) Calculate  $Q_t$  by equation 6 then re-calculate  $K$  by equation 8
  - 7) Get  $K$  reference individuals from  $P$  to population  $P_R$   
*/\* Updating, validating the surrogate model \*/*
  - 8) Classifying the combination of  $P_R$  and  $Arc$  into two categories  $C$  ( $C_1$  and  $C_2$ ) by  $Classify()$  procedure is presented in [18]
  - 9) Rate each solutions in  $C_2$  and store in  $rr$
  - 10) Calculate  $tr = \min\{rr, 1 - rr\}$
  - 11) Partitioning the combination of  $Arc$  and  $C$  into  $D_{train}$  and  $D_{test}$  with  $DataPartition()$  procedure in [18]
  - 12) Training neural network  $net$  by  $Train()$  procedure [18] with  $D_{train}$  and  $T$
  - 13) Validation  $net$  with  $D_{test}$  into  $[p_1, p_2]$   
*/\* Using the surrogate model \*/*
  - 14) Using a surrogate-assisted selection strategy (with  $SASelection(P, P_R, p_1, p_2, g_{max}, tr)$  procedure in [18]) to  $Q$  population
  - 15) Combine the archive  $Arc$  and  $Q$  to  $Arc$
  - 16) Using a radial projection based selection (with  $RadialSelection(P \cup Q, N)$  procedure in [18]) to select  $N$  individuals to  $P$ .
  - 17) Set  $t = t + |Q|$
  - 18) **end while**
- 

aspect of performance. Therefore, it could happen that the set of solutions is very close to the Pareto optimal front (POF), but it does not cover the entire of the POF.

The measure IGD takes into account both convergence and spread to all parts of the POS. The first-norm equation

for IGD is as follows:

$$IGD = \frac{\sum_{i=1}^{\overline{N}} \overline{d}_i}{\overline{N}} \quad (10)$$

where  $\overline{d}_i$  is the Euclidean distance (in objective space) from solution  $i$  in the POS to the nearest solution in  $P$ , and  $\overline{N}$  is the size of the POS. In order to get a good value for IGD (ideally zero),  $P$  needs to cover all parts of the POS. However, this method only focuses on the solution that is closest to the solution in the POS indicating that a solution in  $P$  might not take part in this calculation.

### 3. Results and Analysis

#### a) M-K-RVEA

In order to analyze the performance of M-K-RVEA, with maximum of 20000 evaluations in 30 independent runs, we reported the results of GD, IGD measurement in rows ADC on Tables I, II. We also reported and compared the behaviors of K-RVEA (with the default  $w_{max}$  is 30) and M-K-RVEA.

Analyzing and comparing the results, we can found that M-K-RVEA with adaptive parameter control can get more balance of convergence and diversity for obtained population. In details, in the early stages, it gets better in 6 problems on GD and 6 problems on IGD. During the generations (the later stages), M-K-RVEA gets equal or better on 5 problems, one is the same. It is also better in 6 problems on IGD. In the last stages, M-K-RVEA is gets better in 7 problems on GD and 5 problems in IGD. In other problems, K-RVEA gets the same results with M-K-RVEA or a bit better than M-K-RVEA.

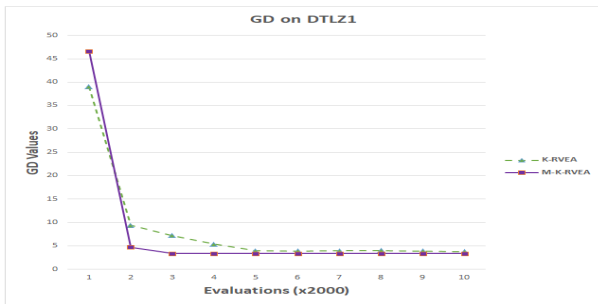


Figure 11. The behavior of M-K-RVEA and K-RVEA on DTLZ1 (GD)

In general, through the experimental results, it is proposed to use adaptive parameter control that has an effective effect to implement the optimal target optimization algorithm using general surrogate functions, algorithms based on Kriging model in particular. The obvious meaning of using the response parameter is to assess the quality of the population at the time of determining whether

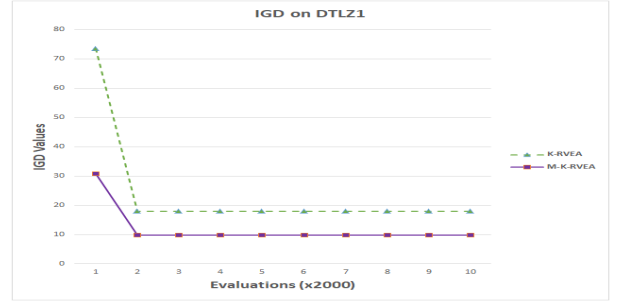


Figure 12. The behavior of M-K-RVEA and K-RVEA on DTLZ1 (IGD)

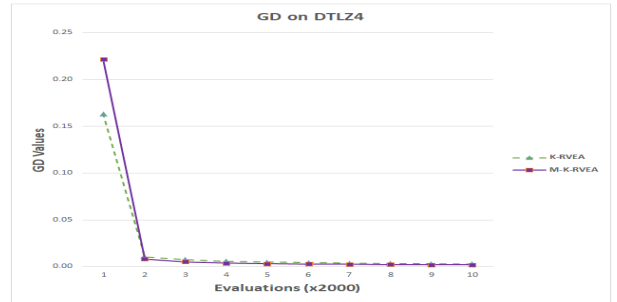


Figure 13. The behavior of M-K-RVEA and K-RVEA on DTLZ4 (GD)

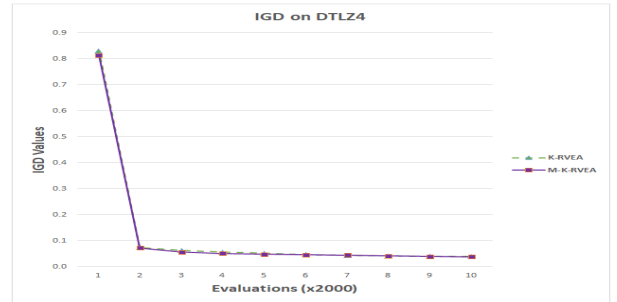


Figure 14. The behavior of M-K-RVEA and K-RVEA on DTLZ4 (IGD)

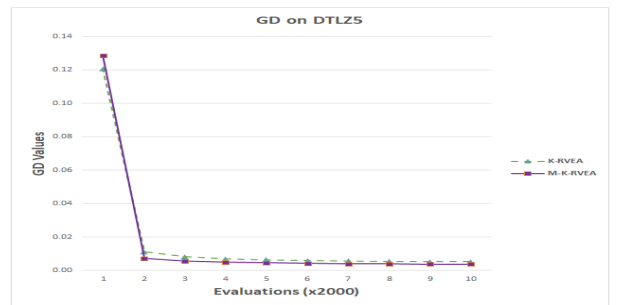


Figure 15. The behavior of M-K-RVEA and K-RVEA on DTLZ5 (GD)

to update the Kriging model or not? The results using adaptive parameter control demonstrate the hypothesis of the impact of the time process, the current convergence quality of the population, and the characteristics of the problem on maintaining the equilibrium of convergence

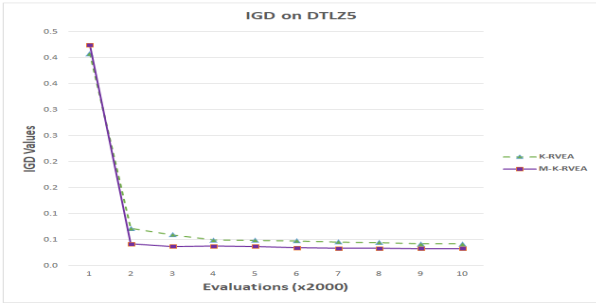


Figure 16. The behavior of M-K-RVEA and K-RVEA on DTLZ5 (IGD)

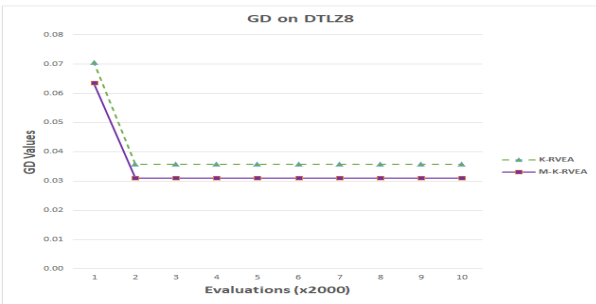


Figure 17. The behavior of M-K-RVEA and K-RVEA on DTLZ8 (GD)

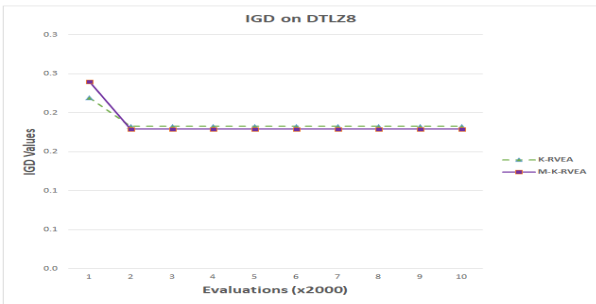


Figure 18. The behavior of M-K-RVEA and K-RVEA on DTLZ8 (IGD)

quality and diversity of the population, through ensuring a balance between the exploration and exploitation process of evolution. The effectiveness of parameter selection in the current stages also depends on the quality and effectiveness of the previous stage, proving that a timely adjustment to maintain a balance between exploration and exploitation is needed to set. However, from the results of some problems, K-RVEA has a better quality, although not large, it also gives us issues that need to be adequately posed such as: stability, sustainability, random factors course, etc. These factors influence the choice of parameters appropriately, or guide evolution to better efficiency, it will reduce noises to improve the robustness of the algorithm.

*b) M-CSEA*

The results for the experiments are shown in Tables IV, V. We also reported and compared the behaviors

of CSEA (with the default  $K$  is 6) and M-CSEA, some of screenshots are shown in Figures: 19 to 26.

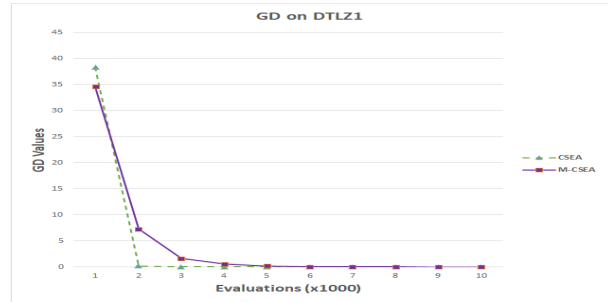


Figure 19. The behavior of M-CSEA and CSEA on DTLZ1 (GD)

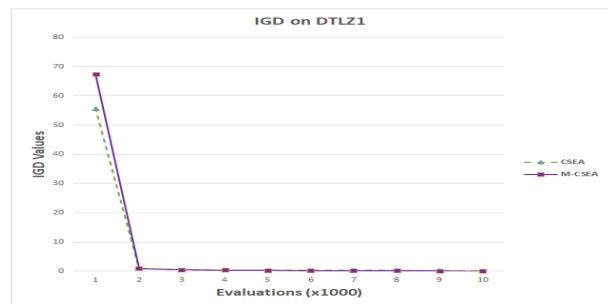


Figure 20. The behavior of M-CSEA and CSEA on DTLZ1 (IGD)

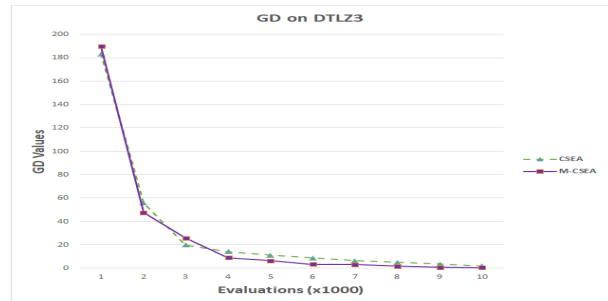


Figure 21. The behavior of M-CSEA and CSEA on DTLZ3(GD)

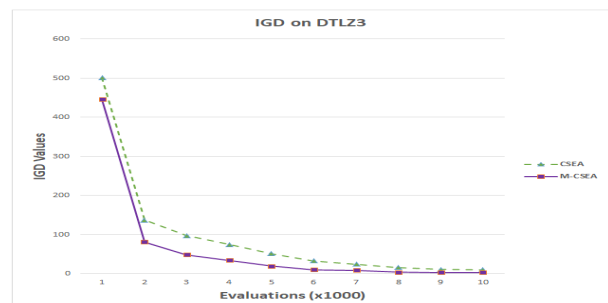


Figure 22. The behavior of M-CSEA and CSEA on DTLZ3 (IGD)

Analysing the results, with maximum of 10000 evaluations in 30 independent runs, in comparing we can found

TABLE IV  
GD VALUES OF CSEA AND M-CSEA ON DTLZS BENCHMARK SET

MOEAs \ Evaluations	1000	2000	3000	4000	5000	6000	7000	8000	9000	10000
DTLZ1										
CSEA	38.304222	0.249517037	0.093476884	0.067478989	0.049841399	0.039018872	0.037277541	0.036718244	0.038506013	0.023741603
M-CSEA	34.55195154	7.293731419	1.659906579	0.593031306	0.253315597	0.093995118	0.049742977	0.045665123	0.007362405	0.004428484
DTLZ2										
CSEA	0.111937965	0.003129364	0.001690321	0.001120471	0.00073779	0.000533606	0.000432797	0.000362262	0.000307509	0.000255133
M-CSEA	0.10952606	0.003268781	0.001166987	0.000817451	0.000578712	0.000431733	0.00034105	0.000305797	0.00026639	0.000231731
DTLZ3										
CSEA	183.1624496	55.90315143	19.85598041	14.28060864	11.19028972	8.806444816	6.52776933	5.115575763	3.579570234	2.017174022
M-CSEA	189.4951725	47.64296364	25.87443973	9.138374728	6.40377985	3.355073622	3.030661077	1.628800101	0.748851809	0.575976429
DTLZ4										
CSEA	0.16005859	0.009324148	0.00158179	0.000911676	0.00062676	0.00050403	0.00043422	0.000366466	0.000306355	0.000280448
M-CSEA	0.14623881	0.00786399	0.002191418	0.001288786	0.000863088	0.000676222	0.00057595	0.000471852	0.000425611	0.00039424
DTLZ5										
CSEA	0.143045187	0.009094896	0.002914526	0.001666095	0.000700365	0.00038713	0.000316341	0.000294842	0.000244925	0.000211265
M-CSEA	0.129456125	0.005469629	0.002643803	0.001694055	0.000970129	0.000418626	0.000343096	0.000282539	0.000252634	0.000234312
DTLZ6										
CSEA	0.937372419	0.490257754	3.61595E-06	1.61995E-06	1.18903E-06	9.76567E-07	8.60556E-07	7.69527E-07	7.12257E-07	6.60703E-07
M-CSEA	0.982408347	0.392787218	0.103280342	0.001726882	0.000270277	0.000137908	1.72537E-05	1.31121E-05	7.19235E-06	5.97902E-06
DTLZ7										
CSEA	2.115699646	0.037620821	0.002642258	0.001528894	0.000967458	0.000684218	0.000527802	0.000431631	0.000384352	0.000343302
M-CSEA	2.631468079	0.051293186	0.008160607	0.00309386	0.001154022	0.000798814	0.000458733	0.000350707	0.000289746	0.000250762
DTLZ8										
CSEA	0.058641647	0.017278221	0.017278221	0.017278221	0.017278221	0.017278221	0.017278221	0.017278221	0.017278221	0.017278221
M-CSEA	0.052658985	0.015295372	0.015295372	0.015295372	0.015295372	0.015295372	0.015295372	0.015295372	0.015295372	0.015295372

TABLE V  
IGD VALUES OF CSEA AND M-CSEA ON DTLZS BENCHMARK SET

MOEAs \ Evaluations	1000	2000	3000	4000	5000	6000	7000	8000	9000	10000
DTLZ1										
CSEA	55.70363999	0.984059887	0.535546843	0.422683497	0.319925917	0.307731215	0.304742746	0.303695285	0.257744343	0.079166362
M-CSEA	67.37843306	0.963184028	0.503859492	0.390551114	0.29077677	0.249641266	0.238247688	0.230690889	0.083235634	0.055616082
DTLZ2										
CSEA	0.477951349	0.110380642	0.074753971	0.065342481	0.057251695	0.052646793	0.050160575	0.047509835	0.045705387	0.043639601
M-CSEA	0.445656061	0.105494378	0.077526559	0.069578473	0.064227686	0.059277889	0.056366055	0.055084377	0.054108945	0.051323836
DTLZ3										
CSEA	500.9795407	137.0799636	97.40436478	74.77410613	51.0949479	32.67368451	24.24581574	15.80437507	10.63840422	9.87021718
M-CSEA	445.3088212	81.03428807	48.27255007	34.01990624	19.12160089	9.809608997	8.919420738	3.96288272	3.06367452	2.820373235
DTLZ4										
CSEA	0.920569575	0.177499921	0.083470208	0.065663179	0.061877807	0.058606665	0.053952637	0.051886509	0.048899923	0.047121413
M-CSEA	0.867610961	0.167420331	0.074209237	0.05987287	0.052081465	0.048150374	0.044544918	0.041799879	0.039348027	0.037201222
DTLZ5										
CSEA	0.479076784	0.049762862	0.019695687	0.012999939	0.008494638	0.006550633	0.005835436	0.005027044	0.004493137	0.004167442
M-CSEA	0.424670314	0.03700648	0.01817932	0.013216065	0.009003176	0.007227901	0.006568781	0.006156097	0.005924713	0.004950872
DTLZ6										
CSEA	8.52857946	1.487209415	0.158712128	0.018593501	0.009388897	0.007934719	0.006343882	0.005608391	0.005024294	0.004639098
M-CSEA	8.547355148	1.926878075	0.117556984	0.015819266	0.00962453	0.006568246	0.004810618	0.003727698	0.003479865	0.003282058
DTLZ7										
CSEA	7.993415905	0.391054762	0.108179002	0.076912183	0.059149384	0.047771425	0.041651639	0.037564026	0.033012235	0.031701786
M-CSEA	8.948192414	1.211191478	0.068162702	0.050790774	0.042785891	0.038499199	0.035339474	0.032621292	0.031239453	0.030098333
DTLZ8										
CSEA	0.24116755	0.136231997	0.136231997	0.136231997	0.136231997	0.136231997	0.136231997	0.136231997	0.136231997	0.136231997
M-CSEA	0.245041877	0.111925885	0.111925885	0.111925885	0.111925885	0.111925885	0.111925885	0.111925885	0.111925885	0.111925885

TABLE VI  
PERFORMANCE TIME FOR K-RVEA AND M-K-RVEA

Problems	K-RVEA (seconds)	M-K-RVEA (seconds)	Faster (%)
DTLZ1	6,864	6,087	11.32
DTLZ2	9,845	9,694	1.53
DTLZ3	15,621	13,851	11.33
DTLZ4	17,432	14,360	17.62
DTLZ5	11,230	10,375	7.61
DTLZ6	19,901	19,866	0.18
DTLZ7	28,823	26,667	7.48
DTLZ8	118,080	105,529	10.63
DTLZ9	34,750	33,969	2.25

TABLE VII  
PERFORMANCE TIME FOR CSEA AND M-CSEA

Problems	K-RVEA (seconds)	M-K-RVEA (seconds)	Faster (%)
DTLZ1	3,952	2,824	28.54
DTLZ2	5,067	5,009	1.14
DTLZ3	12,340	11,337	8.13
DTLZ4	12,53	9,781	21.96
DTLZ5	7,864	6,339	19.39
DTLZ6	13,468	13,166	2.24
DTLZ7	16,446	15,323	6.83
DTLZ8	80,781	77,554	3.99

that, M-CSEA with dynamic selection strategy can get more balance of convergence and diversity for obtained population. In details, in the early stages (since 4000 evaluations), it gets better in DTLZ1, DTLZ2, DTLZ3, DTLZ6, DTLZ8 problems on GD and DTLZ1, DTLZ3, DTLZ4, DTLZ6, DTLZ7, DTLZ8 problems on IGD. During the generations (since 7000 evaluations), M-CSEA gets better in DTLZ2,

DTLZ3, DTLZ6, DTLZ7, DTLZ8 on GD. It is also better in DTLZ1, DTLZ3, DTLZ4, DTLZ6, DTLZ7, DTLZ8 on IGD. In the last stages (completed 10000 evaluations), M-CSEA is gets better in DTLZ1, DTLZ2, DTLZ3, DTLZ5, DTLZ6, DTLZ7, DTLZ8 on GD and DTLZ1, DTLZ2, DTLZ3, DTLZ6, DTLZ7, DTLZ8 on IGD. In other problems, CSEA gets a bit better than M-CSEA.

Besides, in terms of performance time, on the test

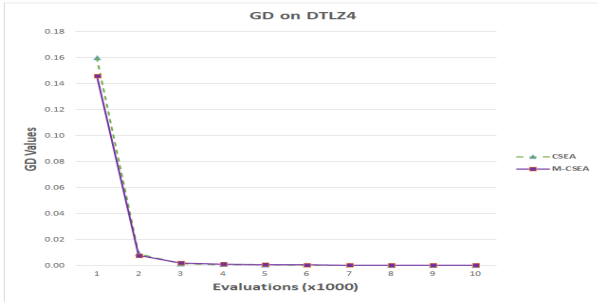


Figure 23. The behavior of M-CSEA and CSEA on DTLZ4 (GD)

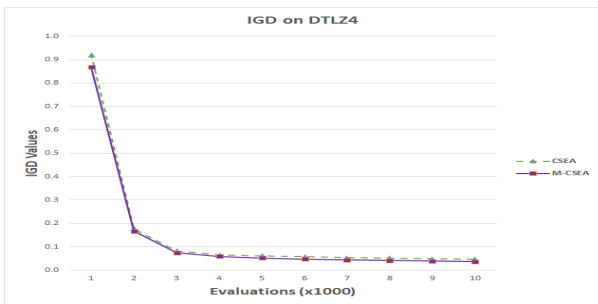


Figure 24. The behavior of M-CSEA and CSEA on DTLZ4 (IGD)

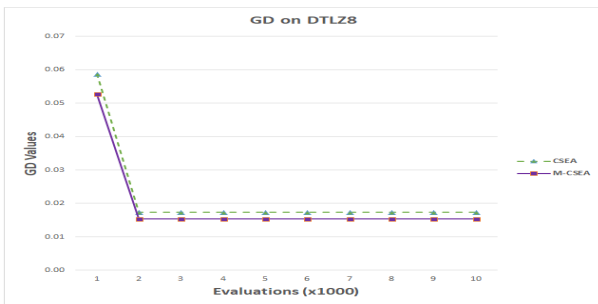


Figure 25. The behavior of M-CSEA and CSEA on DTLZ8 (GD)

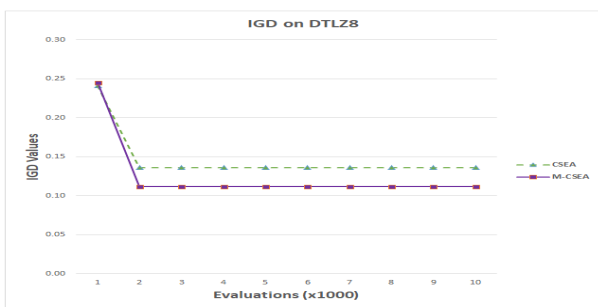


Figure 26. The behavior of M-CSEA and CSEA on DTLZ8 (IGD)

environment, HP ProLiant DL380 with CPU: 2xIntel Xeon Silver 4110 Processor 11M cache 2.10GHz, RAM: 4x31GB PC4-21300, HDD: 4x1TB 7200 rpm SATA 6Gb/s, run-time recognition result between K-RVEA and M-K-RVEA; CSEA and M-CSEA are shown in Table VI and VII.

From the mechanism of applying the guidance method in M-K-RVEA and M-CSEA we can find that, M-K-RVEA with adaptive guidance mechanism, in the early stages, the number of using the origin function increases because of the need to increase convergence, however, when the evolution time is long enough, the training dataset or the external archive is good quality, the frequency of which is mainly used by the surrogate function. On the other hand, the original function is complex, the cost is much larger than the surrogate function, so with a reasonable range of variation in the proposed paper, the overall calculation time will be significantly reduced. This depends on the complexity of the original problem. The M-K-RVEA example greatly reduces time in DTLZ1, DTLZ3, DTLZ4 and DTLZ8 problems.

With the M-CSEA, with an adaptive guidance mechanism in determining the number of samples for model training, in the first phase, it is necessary to use a lot of data from the population to train the model, however with the evolution time is long enough, the number of samples needed is reduced, the search process will be faster, due to the reduced number of samples used. M-CSEA is faster on DTLZ1, DTLZ4, DTLZ5 problems. Here, the different performance efficiency between M-K-RVEA and M-CSEA shows that in addition to the impact assessment raised factors, there are other influencing factors from the techniques used in search, selection, crossover and mutation of each algorithm.

Overall assessment, in addition to a significant improvement in the convergence quality and diversity of the obtained solutions as well as maintaining a balance between exploitation and exploration of the evolutionary process, the proposed guidance method also helps to reduce the computation time of surrogate assisted multi-objective algorithms, especially suitable for solving expensive problems.

In general, through the experimental results, it is proposed to use new selection strategy with adaptive number of reference solutions, which has an effect to implement the optimal target optimization algorithm using general surrogate functions, selection of reference solutions for sampling in machine learning in the FNNs. The obvious meaning of using the response parameter is to assess the quality of the population to determine how many reference solutions are needed for forming the boundary of the classification. The results using dynamic selection strategy by the  $K$  parameter demonstrate the hypothesis of the impact of the time process, the current convergence quality of the population, and the characteristics of the problem on maintaining the equilibrium of convergence and diversity quality of the population, through ensuring a balance between the exploration and exploitation process

of evolution. The effectiveness of adaptive selection in the current stages also depends on the quality and effectiveness of the previous stage, proving that a timely adjustment to maintain a balance between exploration and exploitation is needed. However, from the results of some problems, CSEA has a better quality, although not large, it also gives us issues that need to be adequately posed such as: stability, sustainability, random factors course, etc. Those factors influence the choice of parameters appropriately, or guide evolution to better efficiency, the robustness of the algorithm will be improved by reducing the noises with the dynamic selection strategy.

## VI. CONCLUSIONS

Using surrogate model is an advanced and effective technique in solving expensive problems with MOEAs. However, on the principle of using the surrogate function to replace the original objective function with its own techniques, although it makes the algorithm flexible and diverse, it also makes the evolutionary environment more turbulent, impact the power of the algorithm. This is a research problem set out in order to improve the quality of algorithms using surrogate models, creating effective algorithms that are comprehensive, capable of applying to problems in practice.

The focus of study is on the effects of noisy environments on the performance aspects of MOEAs using surrogate model, the paper has analyzed and evaluated the effects of techniques used in MOEAs using surrogate models that are Kriging, FNNs and guidance techniques. The paper has also analyzed and evaluated factors in parameter selection about frequency of using surrogate model, sample selection methods for training models, the binding factors between sustainability and convergence and diversity through the values of the GD, IGD measures over time of evolution. The paper also proposes to use instructional techniques to automatically adjust algorithms to maintain the convergence and diversity properties as well as the exploration and exploitation capabilities of the evolutionary process. Through experiments on K-RVEA, CSEA and the modified versions, the results showed that with proposed guidance techniques, the algorithms have been improved on quality and thereby reducing noise for the evolutionary environment, contributing to improve the robustness, creating more effective and comprehensive innovative algorithms.

In the next studies, we will study the impacts of practical problems in design, decision making... combined with the above research results to adjust and supplement flexible guidance techniques, responding more effectively to solving expensive problems in practice.

## REFERENCES

- [1] G. Matheron, "Principles of geostatistics," *Economic geology*, vol. 58, no. 8, pp. 1246–1266, 1963.
- [2] S. S. Garud, I. A. Karimi, and M. Kraft, "Design of computer experiments: A review," *Computers & Chemical Engineering*, vol. 106, pp. 71–95, 2017.
- [3] F. Bittner and I. Hahn, "Kriging-assisted multi-objective particle swarm optimization of permanent magnet synchronous machine for hybrid and electric cars," in *Electric Machines & Drives Conference (IEMDC), 2013 IEEE International*. IEEE, 2013, pp. 15–22.
- [4] S. Choi, J. J. Alonso, and H. S. Chung, "Design of a low-boom supersonic business jet using evolutionary algorithms and an adaptive unstructured mesh method," *AIAA paper*, vol. 1758, p. 2004, 2004.
- [5] M. T. Emmerich, K. C. Giannakoglou, and B. Naujoks, "Single-and multiobjective evolutionary optimization assisted by gaussian random field metamodelling," *IEEE Transactions on Evolutionary Computation*, vol. 10, no. 4, pp. 421–439, 2006.
- [6] L. Gonzalez, J. Periaux, K. Srinivas, and E. Whitney, "A generic framework for the design optimisation of multidisciplinary uav intelligent systems using evolutionary computing," in *44th AIAA Aerospace Sciences Meeting and Exhibit*, 2006, p. 1475.
- [7] A. Husain and K.-Y. Kim, "Enhanced multi-objective optimization of a microchannel heat sink through evolutionary algorithm coupled with multiple surrogate models," *Applied Thermal Engineering*, vol. 30, no. 13, pp. 1683–1691, 2010.
- [8] I. Voutchkov and A. Keane, "Multi-objective optimization using surrogates," in *Computational Intelligence in Optimization*. Springer, 2010, pp. 155–175.
- [9] Q. Zhang, W. Liu, E. Tsang, and B. Virginas, "Expensive multiobjective optimization by moea/d with gaussian process model," *IEEE Transactions on Evolutionary Computation*, vol. 14, no. 3, pp. 456–474, 2010.
- [10] S. Maroufpoor, O. Bozorg-Haddad, and X. Chu, "Geostatistics: principles and methods," in *Handbook of Probabilistic Models*. Elsevier, 2020, pp. 229–242.
- [11] K. Deb and H. Gupta, "Introducing robustness in multi-objective optimization," *Evolutionary computation*, vol. 14, no. 4, pp. 463–494, 2006.
- [12] A. Gaspar-Cunha and J. A. Covas, "Robustness in multi-objective optimization using evolutionary algorithms," *Computational Optimization and Applications*, vol. 39, no. 1, pp. 75–96, 2008.
- [13] S. Gunawan and S. Azarm, "Multi-objective robust optimization using a sensitivity region concept," *Structural and Multidisciplinary Optimization*, vol. 29, no. 1, pp. 50–60, 2005.
- [14] M. Dellnitz and K. Witting, "Computation of robust pareto points," *International Journal of Computing Science and Mathematics*, vol. 2, no. 3, pp. 243–266, 2009.
- [15] L. T. Bui, D. Essam, H. A. Abbass, and D. Green, "Performance analysis of evolutionary multi-objective optimization methods in noisy environments," in *Proceedings of the 8th Asia Pacific symposium on intelligent and evolutionary systems*. Citeseer, 2004, pp. 29–39.
- [16] E. Zitzler, L. Thiele, and K. Deb, "Comparison of multi-objective evolutionary algorithms: Empirical results," *Evolutionary Computation*, vol. 8, no. 1, pp. 173–195, 2000.
- [17] T. Chugh, Y. Jin, K. Miettinen, J. Hakanen, and K. Sindhya, "K-rvea: a kriging-assisted evolutionary algorithm for many-objective optimization," *Reports of the Department of Mathematical Information Technology, Series B, Scientific Computing no. B*, vol. 2, 2016.
- [18] 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 Transactions on Evolutionary Computation*, vol. 23, no. 1, pp. 74–88, 2018.

- [19] R. Cheng, Y. Jin, M. Olhofer, and B. Sendhoff, “A reference vector guided evolutionary algorithm for many-objective optimization,” *IEEE Transactions on Evolutionary Computation*, vol. 20, no. 5, pp. 773–791, 2016.
- [20] K. Deb, L. Thiele, M. Laumanns, and E. Zitzler, “Scalable test problems for evolutionary multi-objective optimization, TIK-Report no. 112,” *Computer Engineering and Networks Laboratory (TIK), Swiss Federal Institute of Technology (ETH), Zurich*, Tech. Rep., 2001.
- [21] V. L. Vachhani, V. K. Dabhi, and H. B. Prajapati, “Survey of multi objective evolutionary algorithms,” in *2015 International Conference on Circuits, Power and Computing Technologies [ICCPCT-2015]*. IEEE, 2015, pp. 1–9.



**Hoai Xuan NGUYEN** received the Ph.D. degree in computer science from the University of New South Wales, Kensington, Australia. He has been involved with academics including teaching and research since 1998. Currently, he is an Associate Professor and Director of Vietnam Institute of Artificial Intelligence, Hanoi, Vietnam.

He is researching in the field of Intelligence Computation, Natural language processing; Operations research.

Email: nxhoai@gmail.com.



**Dinh Duc NGUYEN** has his B.Sc (2000) and M.Tech (2010) degree from Le Quy Don Technical University. Currently he is a PhD Candidate at the Military Information Technology Institute, Academy of Military Science and Technology. His research interests are in the areas of Evolutionary Multi-objective Optimizations, Computational Intelligence, Multi-agent systems, Simulation systems, UAV and their applications to defence and security domain.

Email: nddinh76@gmail.com.



**Long NGUYEN** received his PhD in 2015 in Fundamentals of Mathematics for Information Technology from Le Quy Don Technical University. Currently he is a Associate Professor at the Department of Information Technology, National Defense Academy. His research interests are in the areas of Evolutionary Multi-objective Optimizations, Computational Intelligence, Multi-agent systems, Cyber warfare, Artificial Intelligence and their applications to defence and security domain.

Email: longit76@gmail.com.