MOEA/D-S^3: MOEA/D using SVM-based Surrogates adjusted to Subproblems for Many objective optimization

Takumi Sonoda
Collage of Engineering Science
Yokohama National University
Yokohama, Japan
sonoda-takumi-gp@ynu.jp

Masaya Nakata
Collage of Engineering Science
Yokohama National University
Yokohama, Japan
nakata-masaya-tb@ynu.ac.jp

***Abstract*—This paper proposes a surrogate-assisted MOEA/D using SVM-based surrogates adjusted to subproblems (MOEA/D-S^3), which intends to achieve the following technical advantages. Firstly, in order to construct a proper surrogate while reducing learning cost to construct surrogates, a surrogate is an SVM-classifier that identifies a specific region of good solutions and thus its learning cost should be lower than a popular alternative approach, i.e., fitness approximation. Secondly, relying on the first advantage, multiple surrogates are constructed and each surrogate, like an expert, is adjusted to each subproblem defined in the MOEA/D framework in order to improve diversity and convergence of the Pareto set. Experimental results show that MOEA/D-S^3 outperforms MOEA/D on a number of many-objective benchmark problems.**

***Index Terms*—surrogate-assisted evolutionary algorithm, many-objective optimization**

I. INTRODUCTION

Real-world multi-objective optimization problems often belong to a computationally expensive problem in evaluating the fitness of solutions. For those problems, surrogate-assisted evolutionary algorithms (SAEAs) [1] can be a promised approach and many versions have been introduced in the literature. As pointed out in [2], while some fundamental challenges on SAEAs need to be further explored, e.g. model selection and model-update frequency, a recent challenge is to develop effective SAEAs for many-objective optimization problems having more than three objective functions.

A basic idea of SAEAs is to estimate good solutions by utilizing a surrogate in order to reduce expensive fitness evaluations. While various functionalities of surrogate have been proposed, e.g. fitness approximation [3] and constraint estimation [4], we are interested in the *fitness replacement* approach [5], i.e., estimation of good solutions. Since machine learning techniques are often employed to construct those surrogate models, SAEAs are required to efficiently solve a machine learning task in order to construct a proper surrogate model. In fact, in terms of machine learning's insight, the following two difficulties involved in SAEAs should be considered. Firstly, the observation of the objective space is extremely restricted, i.e., a limited number of data samples, to refine the surrogate model. Thus, SAEAs may suffer to construct a proper surrogate model that provides a reliable output to estimate a worth of unevaluated solutions, resulting in a slow convergence towards the true Pareto set. Secondly, the samples, i.e., evaluated solutions, tend to be sparse but also be strongly biased in specific regions of the solution space. Thus, the surrogate model may be undesirably adjusted to the specific regions (i.e., overfitting) dependent on a distribution of the samples, which degrades a diversity of the Pareto set.

Those two difficulties are even more emphasized with the increase of the number of objective functions. This is because that in terms of machine learning's insight, a problem complexity of machine learning problems defined in the SAEA frameworks (e.g, regression), clearly increases dependent on the number of objective functions. In 2018, Pan introduced CSEA as a version of classification-based SAEA [6] handling a lower problem difficulty than fitness approximation; and CSEA successfully outperformed the state-of-the-art of SAEAs e.g. MOEA/D-EGO [7] and K-RVEA [2] on most of many-objective benchmark problems. Interestingly, MOEA/D-EGO, which constructs multiple Kriging models, still outperforms CSEA on few benchmark problems [6]. This paper is inspired by those recent results. Note that the detailed summary of related works is described in Section II.

Consequently, we can suppose the following possible strategies to relax those two difficulties on computationally expensive many-objective optimization problems. For the first difficulty, i.e., the limited number of samples, a low functionality of the surrogate, e.g. classifier like CSEA, can be suitable rather than a rich functionality such as fitness approximation. This intends to reduce a learning cost in order to construct as proper surrogate as possible under the limited samples. For the second difficulty, i.e., the samples biased in the specific regions, multiple surrogate models, like MOEA/D-EGO, may be suitable to improve a diversity of the Pareto set. This intends to improve a diversity of the Pareto set by exploring various specific regions with multiple surrogates.

Accordingly, in this paper, we introduce a version of classification-based SAEA. Our proposed method integrates

978-1-7281-6929-3/20/$31.00 ©2020 IEEE

multiple SVM-based classifiers into MOEA/D [8] and each surrogate is adjusted to each subproblem (i.e., a scalarization function) defined in the MOEA/D framework; and so we call it as MOEA/D using SVM-based surrogates adjusted to subproblems (MOEA/D-S^3). In detail, each SVM-based classifier learns a specific region of good solutions to solve a corresponding subproblem and predicts whether unevaluated solutions exist in its region in order to identify candidates of solutions to be evaluated with actual objective functions.

This paper is organized as follows. Section II describes related works of the SAEAs; and we briefly introduce the mechanisms of MOEA/D and SVM in Section III. In Section IV, we explain the detailed mechanism of MOEA/D-S^3. In Sections V and VI, we show experimental results on many-objective benchmark problems and an additional result as an analysis of hyper-parameters used in MOEA/D-S^3, respectively. Finally, we summarize the contributions of this paper in Section VII. All figures shown in this paper are best viewed in color.

II. Related work

A basic approach of SAEAs is to construct a surrogate model that approximates objective functions from evaluated solutions. Then, an optimizer minimizes approximated objective functions in order to identify candidates of solutions to be evaluated by actual objective functions. In [5], this approach, e.g. GA-ANN [9], MOEA/D-RBF [10], SAMOEA/SVM [11], is referred as *Direct Fitness Replacement* (DFR). While this approach can be suitable if the systems can have a large number of samples, it is typically hard to construct a reliable surrogate model with low approximation errors on the computationally-expensive many-objective optimization problems. Very few works belonging to the DFR approach, e.g. K-RVEA [2] and HSMEA [4], have been proposed for the many-objective optimization problems.

In recent years, many works proposed different functionalities of surrogate model as an alternative of the approximation model, which is classified to the *Indirect Fitness Replacement* approach (IFR) [5]. For instance, ParEGO [12] converts a multi-objective optimization problem into a single-objective function using a scalarization function; and it constructs a response surface of the scalarization function by the Kriging model [13]. As an extension of ParEGO, MOEA/D-EGO [7] solves multiple subproblems with scalarization functions defined in the MOEA/D framework; MOEA/D-EGO constructs multiple surrogates for the subproblems. Hence, MOEA/D-EGO may be suitable to improve a diversity of solutions since the multiple surrogates are adjusted to the subproblems, i.e., to search various specific regions. As a different approach of IFR, PARETO-SVM [14] and CSEA [6], which are dominance-based approach like NSGA-II [15], are designed to construct a single classifier that identifies good solutions belonging to the Pareto set. A classification task can be defined as having a lower learning cost and thus be more scalable for the increase of the number of objective functions than approximation and regression tasks.

As mentioned in Section I, MOEA/D-S^3 is designed to improve a diversity of solutions by adjusting multiple surrogates to the subproblems, like MOEA/D-EGO; and to reduce the learning cost by constructing a classifier-based surrogate, like PARETO-SVM and CSEA.

III. Brief descriptions of MOEA/D and SVM

A. MOEA/D

MOEA/D is a decomposition-based evolutionary multi-objective algorithm, and a multi-objective optimization problem is divided into N subproblems by a scalarization function. For instance, the Tchebycheff function as a scalarization function, which is employed in this paper, is given by;

$$g(\boldsymbol{x}|\boldsymbol{\lambda}, \boldsymbol{z}^*) = \max_{1\leq i\leq m} \left\{\lambda_i |f_i(\boldsymbol{x}) - z_i^*|\right\}, \tag{1}$$

where $\boldsymbol{\lambda}$ is the weight vector assigned to each subproblem, and $\lambda_i \geq 0, (i = 1, \cdots, m)$ such that $\sum_{i=1}^{m} \lambda_i = 1$; $\boldsymbol{z}^*$ is a set of reference points used for determining the search direction; in a minimization problem, z_i^* is set to the minimum value of each objective function as $z_i^* = \min\left\{f_i(\boldsymbol{x})|\boldsymbol{x} \in S\right\}$ $(i = 1, \cdots, m)$. In fact, this minimum value is not given in advance and so the provisional minimum value $z_i = \min\left\{f_i(\boldsymbol{x})|\boldsymbol{x} \in \Omega\right\}$ is set to the reference point, where Ω is a set of individuals explored so far. Then, each subproblem has the neighborhood, which are subproblems determined by T closest weight vectors. To generate a solution for the i-th subproblem, a crossover is performed to individuals of the i-th subproblem's neighborhood, under an assumption that the subproblems with similar weight vectors have similar regions of optimal solutions. Those processes are summarized in Algorithm 1. Note that Algorithm 1 describes the framework of MOEA/D-DE [16], which will be used in this paper. MOEA/D-DE employs the differential evolution algorithm as a crossover operator. In detail, each element y_k of offspring $\boldsymbol{y}$ for the i-th subproblem is generated by the DE's operator and polynomial mutation;

$$\bar{y}_k = \begin{cases} x_k^i + F \times (x_k^{r_2} - x_k^{r_3}) & \text{with probability } CR, \\ x_k^i & \text{otherwise,} \end{cases} \tag{2}$$

where x_k^i is the k-th element of $\boldsymbol{x}^i$ for the i-th subproblem; r_2 and r_3 are randomly selected from $\boldsymbol{P}$ [16]; F and CR are a scaling factor and a crossover rate, respectively. Then, based on $\bar{y}_k$, y_k is decided by polynomial mutation;

$$y_k = \begin{cases} \bar{y}_k + \sigma_k \times (b_k - a_k) & \text{with probability } p_m, \\ \bar{y}_k & \text{otherwise,} \end{cases} \tag{3}$$

where a_k and b_k are the lower and upper bounds of the k-th decision variable, respectively; p_m is a mutation probability. Here, σ_k is further given by;

$$\sigma_k = \begin{cases} (2 \times rand)^{1/(\eta+1)} - 1 & \text{if } rand < 0.5, \\ 1 - (2 - 2 \times rand)^{1/(\eta+1)} & \text{otherwise,} \end{cases} \tag{4}$$

where η is a distribution index of polynomial mutation. After those operations are conducted, a new offspring $\boldsymbol{y}$ is evaluated and then, used to update the population. Then, the number of

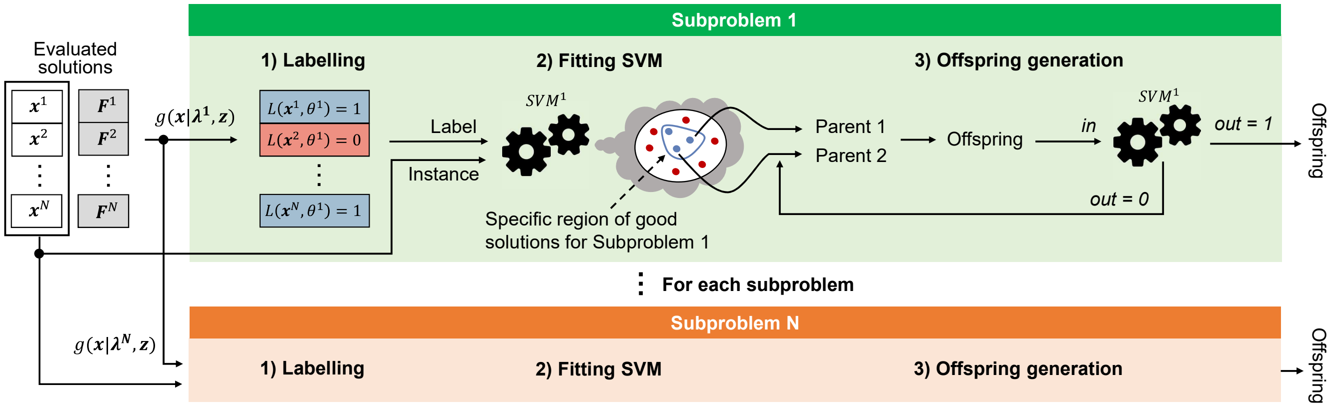

Fig. 1. Surrogate construction and its utilization in MOEA/D-S^3

Algorithm 1 MOEA/D-DE

$\boldsymbol{B}(i) \leftarrow$ Set indexes of i-th subproblem's neighborhood for each $i = 1, \cdots, N$
Generate a population $\mathcal{P} = \{\boldsymbol{x_1}, \cdots, \boldsymbol{x_N}\}$
Initialize $z_j = \min\{f_j(\boldsymbol{x}) | \boldsymbol{x} \in \mathcal{P}\}$ for each $j = 1, \cdots, m$
while terminate criterion is not satisfied **do**
  for $i = 1$ to N **do**
    if $rand < \delta$ **then**
      $\boldsymbol{P} = \boldsymbol{B}(i)$
    else
      $\boldsymbol{P} = \{1, \cdots, N\}$
    end if
    $\boldsymbol{y} \leftarrow$ Generate offspring with parent set $\boldsymbol{P}$
    Evaluate $\boldsymbol{y}$
    Update $z_j = \min\{f_j(\boldsymbol{x}) | \boldsymbol{x} \in \mathcal{P}\}$ for each $j = 1, \cdots, m$
    $c = 0$
    while $c \leq n_r$ or $|\boldsymbol{P}| \neq 0$ **do**
      Pick an index j from $\boldsymbol{P}$
      if $g(\boldsymbol{y} \mid \boldsymbol{\lambda}^i, \boldsymbol{z}) \leq g(\boldsymbol{x}^j \mid \boldsymbol{\lambda}^i, \boldsymbol{z})$ **then**
        $\boldsymbol{x}^j \leftarrow \boldsymbol{y}$ and $c \leftarrow c + 1$
      end if
      Remove an index j from $\boldsymbol{P}$
    end while
  end for
end while
Output $\mathcal{P}$

solutions updated by $\boldsymbol{y}$ is restricted by n_r in MOEA/D-DE. In addition, in Algorithm 1, $\mathcal{P}$ is the population and $\boldsymbol{B}(i)$ is a set of indexes to refer the neighborhood of the i-th subproblem; the neighborhood is selected as $\boldsymbol{P}$ with a probability δ.

B. SVM

The support vector machine SVM [17] is a pattern recognition technique of the supervised learning scheme and is frequently used as a classifier for binary-class classification tasks. In SVM, a decision boundary of classes is trained so as to maximize the summation of the distances between the decision boundary and given training data of each class. In this paper, we use a nonlinear SVM with the RBF kernel given by the following equation;

$$K(\boldsymbol{x}_i, \boldsymbol{x}_j) = \exp(-\gamma \|\boldsymbol{x}_i - \boldsymbol{x}_j\|) \quad (5)$$

where $\gamma > 0$ controls a complexity of the decision boundary; the boundary tends to be a complex shape with the increase of γ. In addition, SVM involves a regularization coefficient C, which controls a tolerance of mis-classification.

IV. MOEA/D-S^3

This section describes the detailed motivation of MOEA/D-S^3 and then its mechanism. As noted in Section III, we use MOEA/D-DE in order to boost the potential performance of MOEA/D-S^3, but our extensions are related only to the MOEA/D framework and so we simply name MOEA/D-S^3.

A. Motivation

MOEA/D-S^3 is inspired by the recent success of CSEA and MOEA/D-EGO. In detail, like CSEA, MOEA/D-S^3 employs a classifier-based surrogate, which is constructed by SVM, in order to identify candidates of solutions worth to be evaluated with the actual objective functions. Like MOEA/D-EGO, it constructs multiple surrogates adjusted to the subproblems defined in the MOEA/D framework.

In addition, each surrogate, like an expert, can be adjusted to each subproblem; suppose objective functions include heterogeneous dependencies between variables, we can expect that like a divide-and-conquer strategy, each surrogate captures an important dependency existed in its subproblem. Besides, with the multiple surrogate scheme on MOEA/D, we can prepare various training samples by converting the evaluated solutions with the different scalarization functions. Consequently, we can improve the diversity of surrogate models to explore multiple specific regions. This can be one of main advantages of MOEA/D-S^3; CSEA constructs one surrogate model that classifiers solutions likely to be in the Pareto set.

B. Mechanism

Fig. 1 shows the main mechanisms of MOEA/D-S^3. As shown in this figure, MOEA/D-S^3 consists of the following two main mechanisms; *surrogate construction*, which constructs the SVM-based classifiers for corresponding subproblems, and *surrogate utilization*, which generates solutions

Algorithm 2 *SVM-Construction*$(\mathcal{P})$

```
1: for i = 1 to N do
2:     G^i = {g(x | λ^i, z) | x ∈ P}
3:     θ^i ← Calculate median value of G^i
4:     L^i = {L(x, θ^i) | x ∈ P}
5:     SVM^i ← Fit SVM model using inputs x ∈ P and labels L^i
6: end for
```

Algorithm 3 *Offspring Generation*$(i, \boldsymbol{P})$

```
1: r = 0
2: y ← Generate offspring with parent set P
3: while Predict(y, SVM^i) ≠ 1 or r < R_max do
4:     y ← Re-generate offspring with parent set P
5:     r ← r + 1
6: end while
```

suggested by the surrogates. Then, those two mechanism are added to the MOEA/D-DE framework.

1) Surrogate construction: MOEA/D-S^3 constructs N SVM-based surrogates corresponding to N subproblems defined in MOEA/D. Here, i-th surrogate, denoted by *SVM*^{i}, is adjusted to the i-th subproblem with the scalarization function $g(\boldsymbol{x} \mid \boldsymbol{\lambda}^i, \boldsymbol{z})$. In MOEA/D-S^3, we formalize a binary-class classification task, which aims to classify unevaluated solutions to either having a higher value of $g(\boldsymbol{x} \mid \boldsymbol{\lambda}^i, \boldsymbol{z})$ or having a lower value than a defined threshold. In other words, the surrogate learns a specific region of good solutions defined by its threshold. The detailed algorithm to construct the surrogate for each subproblem is described in Algorithm 2.

After the initial process of MOEA/D (or MOEA/D-DE) is conducted, for each subproblem, MOEA/D-S^3 produces the training dataset to construct *SVM*^{i}. Here, in order to deal with the classification task, we define and calculate labels of evaluated solutions. The training dataset for the i-th subproblem consists of the population i.e., the set of decision variables $\boldsymbol{x}$; and a sef of labels corresponding to $\boldsymbol{x}$, denoted by $\mathcal{L}^i$. In detail, $\mathcal{L}^i$ is determined with the following procedures. Firstly, to convert the evaluated solutions to the dataset specified for the i-th subproblem, we calculate a value of $g(\boldsymbol{x} \mid \boldsymbol{\lambda}^i, \boldsymbol{z})$ for each $\boldsymbol{x} \in \mathcal{P}$, forming a set of $g(\boldsymbol{x} \mid \boldsymbol{\lambda}^i, \boldsymbol{z})$ denoted by $\mathcal{G}^i$, given by;

$$\mathcal{G}^i = \{g(\boldsymbol{x} \mid \boldsymbol{\lambda}^i, \boldsymbol{z}) \mid \boldsymbol{x} \in \mathcal{P}\}. \tag{6}$$

Next, we calculate the labels based on $\mathcal{G}^i$. In this paper, we simply define *good* solutions $\boldsymbol{x}$ for the i-th subproblem as having a higher value of $g(\boldsymbol{x} \mid \boldsymbol{\lambda}^i, \boldsymbol{z})$ than a threshold; and we here set a threshold θ^i for the i-th subproblem to the median value of $\mathcal{G}^i$. Hence, this threshold can be dynamically tuned dependent on $\mathcal{G}^i$. Note that, while a determination of θ^i should be further explored, we have the following reasons to employ the median value. Firstly, we can avoid a problematic class-imbalanced issue since the number of each class (i.e., label) can be evenly defined. Secondly, we can expect that the local search and the global search in the solution space can be balanced; otherwise SVM would learn a very specific region of good solutions. Then, we calculate $\mathcal{L}^i$ as;

$$\mathcal{L}^i = \{L(\boldsymbol{x}, \theta^i) \mid \boldsymbol{x} \in \mathcal{P}\}, \tag{7}$$

where $L(\boldsymbol{x}, \theta^i)$ returns a binary label of $\boldsymbol{x}$ as;

$$L(\boldsymbol{x}, \theta^i) = \begin{cases} 1 & \text{if } g(\boldsymbol{x} \mid \boldsymbol{\lambda}^i, \boldsymbol{z}) \leq \theta^i, \\ 0 & \text{otherwise.} \end{cases} \tag{8}$$

Finally, MOEA/D-S^3 fits the SVM classifier so that *SVM*^{i} can classify the inputs $\boldsymbol{x} \in \mathcal{P}$ to its correct label $L(\boldsymbol{x}, \theta^i) \in \mathcal{L}^i$. Note that in MOEA/D-S^3, the SVM hyper-parameter γ (see Equation (5)) controls the complexity of the shape of the decision boundary. However, it is unclear how this complexity affects the performance of MOEA/D-S^3; and so we will investigate the impact of γ in Section VI.

2) Surrogate utilization: MOEA/D-S^3 utilizes the surrogate to identify candidates of offspring worth to be evaluated with the actual objective functions. Specifically, we defined that the candidate solutions must exist in the specific region learned by the surrogate. Accordingly, for the i-th subproblem, MOEA/D-S^3 repeats the offspring-generation process until it generates the offspring identified as existing in the specific region, i.e., as having the label "1" (see Equation (8)). To avoid an infinite loop in our implementation, we employ the termination to finalize the offspring-generation process; when the number of regenerations is larger than R_{max}, we accept the latest to be evaluated (even when its offspring does not have the label "1"). However, this is an exceptional case, as MOEA/D-S^3 does not meet this termination in the experimental results reported in Section V. This algorithm is described in Algorithm 3. For the i-th subproblem, the offspring $\boldsymbol{y}$ is generated based on the parent's index set $\boldsymbol{P}$ by the same offspring-generation process as in MOEA/D-DE. Then, *SVM*^{i} predicts the class of $\boldsymbol{y}$, represented by *Predict*($\boldsymbol{y}$, *SVM*^{i}); and MOEA/D-S^3 repeats to generate $\boldsymbol{y}$ until the termination is met.

In addition, we here introduce an optional modification of the parent-selection process to build the parent's indexes $\boldsymbol{P} = \{P_1, P_2, \cdots, P_{|\boldsymbol{P}|}\}$, which intends to enhance the local search applied to the specific regions learned by SVM, i.e, an evolutionary propagation of good solutions having the label "1". In detail, as described in Algorithm 4, for the i-th subproblem, *SVM*^{i} predicts a label of the P_j-th parent solution $\boldsymbol{x}^{P_j}$. Then, we further add P_j to a modified index set $\boldsymbol{P'}$ if $\boldsymbol{x}^{P_j}$ is predicted as having the label "1". Note that, to apply the offspring-generation process of MOEA/D-DE, $|\boldsymbol{P'}|$ must be at least two; and so we randomly select and add unselected indexes to $\boldsymbol{P'}$ if $|\boldsymbol{P'}| < 2$. Then, the offspring-generation process described in Algorithm 3 is performed with $\boldsymbol{P'}$ instead of $\boldsymbol{P}$; and thus Algorithm 4 can be inserted next to the first line (i.e.,, before calculating $\boldsymbol{y}$) of Algorithm 2.

As the overall procedure, MOEA/D-S^3 can be implemented by adding Algorithm 2 next to 3th and 22th lines in Algorithm 1 , respectively; and replacing the line 11 with Algorithm 3.

V. Experiment

In this section, we test MOEA/D-S^3 on many-objective optimization problems, i.e., the WFG test suite [18]. In the WFG problems including k position variables and l distance

Algorithm 4 *Parent selection*$(i, \boldsymbol{P})$

$\boldsymbol{P}' = \{\}$
for $j = 1$ to $|\boldsymbol{P}|$ **do**
  if *Predict*$(\boldsymbol{x}^{P_j}$, *SVM*$^i) = 1$ **then**
    Add P_j to $\boldsymbol{P}'$
  end if
end for
while $|\boldsymbol{P}'| < 2$ **do**
  $P_r \leftarrow$ Select unselected $P_r \in \boldsymbol{P}$ randomly
  Add P_r to $\boldsymbol{P}'$
end while

variables, the previous work recommends that the problem dimension n can be tuned dependent on the number of objective functions m; $k = m-1$, $l = 10$, and $n = k+l$ [18]. Here, we set $m = 4, 6, 8$ as the many-objective problems and so we can get $n = 13, 15, 17$.

A. Experimental setting

We compare MOEA/D-S^3 with MOEA/D-DE. While the state-of-the-art of SAEAs, e.g. CSEA, MOEA/D-EGO have been compared [6], this paper aims to understand the pure effect of our modifications and focuses on its analysis as conducted in the next section. So, we leave an intensive comparison with the alternative SAEAs as future works. We use Hypervolume (HV) [19] as an evaluation criterion to verify the performance of the optimizer; we use the DEAP [20] package to calculate the HV score. For each problem, we repeat 30 trials with different random seeds and then we report median values of 30 trials. We also apply the Wilcoxon signed rank sum test to find the significant difference. The reference point of HV is set to $\{r_1, \cdots, r_m\} = \{11\}^m$.

We use the following parameter settings; for MOEA/D-DE, $N = 100$, $T = 20$, $\delta = 0.9$, $p_m = 1/n$, $\eta = 20$, $CR = 1.0$ and $F = 0.5$; for MOEA/D-S^3, $R_{\max} = 10000$, $\gamma = 1.0$, $C = 100000$ and other parameter settings are the same as in MOEA/D-DE. Note that we set $C = 100000$ in order to reduce the mis-classification rate, i.e., to construct an accurate surrogate model; and we use $\gamma = 1.0$ as a default value but we will investigate a dependency of the performance of MOEA/D-S^3 to γ in the next section. In addition, we use our modified parent selection mechanism introduced in Algorithm 4. The maximum number of generations is set to 300 and thus the total fitness evaluations can be 30000. However, as SAEAs aim to reduce fitness evaluations, we also compare the HV scores at 5th and 10th generation, corresponding to 500 and 1000 fitness evaluations, respectively. In addition, we also report the HV scores at 50th and 100th as well as 300th generation to investigate the performance at later generations.

B. Results

Table I reports the median values of the HV scores. We applied the Wilcoxon signed rank sum test to a pair of MOEA/D-S^3 and MOEA/D-DE for each generation and we confirmed that MOEA/D-S^3 significantly outperforms MOEA/D-DE on the WFG1-9 problems for all the generations, i.e., 5th, 10th, 50th, 150th, 300th ($p < 0.001$). Note that MOEA/D-S^3 always generated all offspring having the label "1".

As shown in this table, MOEA/D-S^3 outperforms MOEA/D-DE on most of the employed problems. Importantly, MOEA/D-S^3 improves the HV scores of MOEA/D-DE at 5th and 10th generations i.e., 500 and 1000 fitness evaluations. Hence, MOEA/D-S^3 successfully boosts the performance at early generations; this improvement can be observed when the number of objective functions are increasing to 8. This is an important advantage on the computationally-expensive many-objective optimization problems. In addition, when the generation is further increasing, MOEA/D-S^3 also stably outperforms MOEA/D-DE. For instance, for WFG1 and WFG2 ($m = 8$), the HV scores of MOEA/D-DE at the end of generations, i.e., 300th, are 1.00E+8 and 2.07E+8, respectively; and MOEA/D-S^3 derived the same scores with a half of fitness evaluations required in MOEA/D-DE, i.e., 150th generation.

VI. Analysis

We further provide analytical insights of MOEA/D-S^3. The first analysis is to investigate how a complexity of the shape of decision boundary learned by SVM affects the performance of MOEA/D-S^3. The second analysis is to validate an effect of our modified parent-selection mechanism.

A. Complexity of decision boundary

As explained in Section IV, the hyper-parameter γ controls the complexity of the shape of the decision boundary learned by SVM. As shown in Figure 2-bottom, with a high value of γ, the decision boundary tends to divide the region of the class into multiple specific regions, like small clusters. In contrast, with a low value of γ, the shape of the boundary can be smooth and tends to hold large regions compared with the high value of γ (see Figure 2-top). Hence, to investigate the impact of the complexity of the decision boundary to the performance of MOEA/D-S^3, we here compare the three versions of MOEA/D-S^3 with $\gamma = \{10^{-5}, 1.0, 10^5\}$ on WFG1 ($m = 8$) with the same experimental setting as in Section V.

Fig. 3 reports the median value of the HV scores of the three versions (and MOEA/D-DE as a baseline) over generations. As shown in this figure, MOEA/D-S^3 with $\gamma = 10^5$ derives the best performance compared with the other two versions. While the performance of MOEA/D-S^3 with $\gamma = 10^{-5}$ is better than the case of $\gamma = 1.0$ at the end of generations, we can expect that a complex decision boundary with a high value of γ may promote the local search while improving a diversity. This is because that a high value of γ tends to divide the region of class into multiple specific regions; and then the multiple regions dispersedly exist in the solution space. Accordingly, we further investigate the distributions of explored solutions by those three versions. In detail, we apply the t-SNE [21] method as a non-linear dimension reduction technique ($\mathbb{R}^n \rightarrow \mathbb{R}^2$). Fig. 4 showed the distributions of solutions in the population $\mathcal{P}$ obtained by each version and that of MOEA/D-DE for baseline. At the 2nd generation, all the four versions generate solutions having a similar diversity

TABLE I
MEDIAN VALUES OF HYPERVOLUME OF MOEA/D-S^3 AND MOEA/D-DE, WHICH ARE OBTAINED AT 5TH, 10TH, 50TH, 150TH, 300TH GENERATIONS, RESPECTIVELY. THE p-VALUES CALCULATED FROM THE WILCOXON SIGNED RANK SUM TEST ARE SUMMARIZED AT THE BOTTOM.

		5th generation		10th generation		50th generation		150th generation		300th generation	
problem	m	MOEA/D -S^3(ours)	MOEA/D -DE	MOEA/D -S^3(ours)	MOEA/D -DE	MOEA/D -S^3(ours)	MOEA/D -DE	MOEA/D -S^3(ours)	MOEA/D -DE	MOEA/D -S^3(ours)	MOEA/D -DE
WFG1	4	**8.94E+03**	8.85E+03	**9.18E+03**	9.04E+03	**9.50E+03**	9.42E+03	**9.70E+03**	9.62E+03	**9.84E+03**	9.73E+03
	6	**9.12E+05**	9.05E+05	**9.31E+05**	9.22E+05	**9.71E+05**	9.56E+05	**9.88E+05**	9.77E+05	**1.00E+06**	9.88E+05
	8	**9.29E+07**	9.23E+07	**9.49E+07**	9.37E+07	**9.82E+07**	9.73E+07	**1.00E+08**	9.91E+07	**1.02E+08**	1.00E+08
WFG2	4	8.94E+03	**1.28E+04**	9.18E+03	**1.33E+04**	9.50E+03	**1.41E+04**	9.70E+03	**1.44E+04**	9.84E+03	**1.45E+04**
	6	**1.45E+06**	1.44E+06	**1.54E+06**	1.53E+06	**1.68E+06**	1.66E+06	**1.73E+06**	1.72E+06	**1.75E+06**	1.74E+06
	8	**1.60E+08**	1.59E+08	**1.75E+08**	1.73E+08	**1.98E+08**	1.93E+08	**2.07E+08**	2.03E+08	**2.10E+08**	2.07E+08
WFG3	4	1.15E+04	1.15E+04	1.18E+04	1.18E+04	1.23E+04	1.23E+04	1.25E+04	1.25E+04	**1.27E+04**	1.26E+04
	6	**1.08E+06**	1.07E+06	**1.11E+06**	1.10E+06	**1.17E+06**	1.16E+06	**1.21E+06**	1.19E+06	**1.24E+06**	1.22E+06
	8	**9.20E+07**	9.13E+07	**9.65E+07**	9.53E+07	**1.03E+08**	1.02E+08	**1.07E+08**	1.05E+08	**1.10E+08**	1.08E+08
WFG4	4	**1.19E+04**	1.16E+04	**1.27E+04**	1.24E+04	**1.37E+04**	1.35E+04	**1.40E+04**	1.38E+04	**1.42E+04**	1.40E+04
	6	**8.27E+05**	7.70E+05	**9.82E+05**	9.37E+05	**1.32E+06**	1.21E+06	**1.49E+06**	1.38E+06	**1.58E+06**	1.47E+06
	8	**2.31E+07**	2.15E+07	**3.52E+07**	2.92E+07	**6.53E+07**	5.60E+07	**9.51E+07**	7.55E+07	**1.17E+08**	9.02E+07
WFG5	4	1.18E+04	1.18E+04	1.21E+04	1.21E+04	**1.29E+04**	1.26E+04	**1.36E+04**	1.31E+04	**1.39E+04**	1.35E+04
	6	**1.16E+06**	1.14E+06	**1.26E+06**	1.23E+06	**1.38E+06**	1.35E+06	**1.48E+06**	1.42E+06	**1.57E+06**	1.47E+06
	8	**8.72E+07**	8.25E+07	**1.07E+08**	1.02E+08	**1.34E+08**	1.28E+08	**1.52E+08**	1.43E+08	**1.64E+08**	1.51E+08
WFG6	4	**1.20E+04**	1.19E+04	**1.26E+04**	1.25E+04	**1.35E+04**	1.34E+04	**1.39E+04**	1.38E+04	**1.41E+04**	1.40E+04
	6	**1.06E+06**	1.01E+06	**1.21E+06**	1.15E+06	**1.45E+06**	1.39E+06	**1.59E+06**	1.52E+06	**1.64E+06**	1.60E+06
	8	**5.46E+07**	5.25E+07	**7.01E+07**	6.86E+07	**1.16E+08**	1.08E+08	**1.42E+08**	1.31E+08	**1.59E+08**	1.46E+08
WFG7	4	1.25E+04	1.25E+04	**1.31E+04**	1.29E+04	**1.38E+04**	1.37E+04	**1.41E+04**	1.40E+04	**1.43E+04**	1.42E+04
	6	**1.09E+06**	1.06E+06	**1.25E+06**	1.23E+06	**1.51E+06**	1.45E+06	**1.62E+06**	1.55E+06	**1.65E+06**	1.60E+06
	8	**5.08E+07**	4.83E+07	**7.27E+07**	6.85E+07	**1.26E+08**	1.13E+08	**1.50E+08**	1.38E+08	**1.62E+08**	1.54E+08
WFG8	4	**1.15E+04**	1.14E+04	**1.20E+04**	1.19E+04	**1.29E+04**	1.27E+04	**1.33E+04**	1.31E+04	**1.35E+04**	1.34E+04
	6	**9.25E+05**	9.15E+05	**1.05E+06**	1.02E+06	**1.26E+06**	1.23E+06	**1.35E+06**	1.32E+06	**1.40E+06**	1.37E+06
	8	3.57E+07	**3.85E+07**	**4.96E+07**	4.71E+07	**8.06E+07**	7.63E+07	**9.93E+07**	9.14E+07	**1.09E+08**	1.01E+08
WFG9	4	**1.26E+04**	1.25E+04	**1.31E+04**	1.30E+04	**1.38E+04**	1.37E+04	**1.41E+04**	1.40E+04	**1.42E+04**	1.41E+04
	6	**1.23E+06**	1.17E+06	**1.38E+06**	1.33E+06	**1.56E+06**	1.54E+06	**1.64E+06**	1.61E+06	**1.67E+06**	1.65E+06
	8	**8.46E+07**	7.20E+07	**1.17E+08**	1.02E+08	**1.67E+08**	1.59E+08	**1.84E+08**	1.79E+08	**1.91E+08**	1.88E+08
p value		5.15E-04		3.02E-05		2.21E-05		2.21E-05		1.59E-05	

to each other. However, at the 10th generation, MOEA/D-S^{3}s tend to explore specific regions compared with MOEA/D-DE. This tendency can be more highlighted with the increase of γ. In particular, MOEA/D-S^3 with $\gamma = 10^5$ searched very specific regions, as some solutions are overlapped in almost the same region but also their regions dispersedly exist in the solution space. MOEA/D-S^3 with $\gamma = 10^5$ tends to promote the local search while improving a diversity of solutions. Indeed, the diversity is improved at the 50th generation while exploring the specific regions as the local search. Hence, we can suppose that this is why MOEA/D-S^3 with $\gamma = 10^5$ derives the best performance as shown in Fig. 3. However, it is still unclear for an empirical fact that MOEA/D-S^3 with $\gamma = 10^{-5}$ eventually outperforms MOEA/D-S^3 with $\gamma = 1$ at the end of generations. Our supposition is that the local search may improve the performance in this experimental case.

In summary, when increasing the complexity of the shape of decision boundary with a high value of γ, the local search and the global search can be balanced well. We can consider a risk that MOEA/D-S^3 may fall into the local optima with a less diversity of samples. However, this can be avoided in the MOEA/D-S^3 framework, since it is designed to construct the multiple surrogate models and to use the median value of $g(\boldsymbol{x} \mid \boldsymbol{\lambda}^i, \boldsymbol{z})$ as the threshold. In addition, we can control the bias of the local search by tuning the value of γ. Hence, it would be worth to develop a self adaptation of γ.

B. Analysis on parent selection

Our second analysis is to investigate the effect of our modified parent selection described in Algorithm 4. The aim of our modification is to enhance the local search applied to the specific regions learned by SVM and thus it can be expected to improve the performance of MOEA/D-S^3. Here, we test MOEA/D-S^3 without our modification, i.e., parent solutions are selected with the same procedure as in MOEA/D-DE. We use the WFG1-9 problems with the same experimental settings as in Section V.

Table II summarizes the median values of HV scores of the MOEA/D-S^3 without our modified parent-selection. Note that in this table, "+", "$-$" and "$\sim$" represent that MOEA/D-S^3 without the modification derives better, worse, and competitive performances than (or to) MOEA/D-S^3 (with our modification), respectively (see Table I). Hence, we can confirm the effect of our modification when a large number of "$-$" can be observed. For 50th and 300th generations, our modification successfully enables MOEA/D-S^3 to improve the performance of MOEA/D-S^3 without our modification ($p < 0.05$). This confirms the effect of our modification that it promotes a local search around good parents. However, at the 2nd generation, MOEA/D-S^3 without our modification outperforms MOEA/D-S^3 on 15 experimental cases. This fact indicates a drawback of our modification. In detail, our modification may degrade a diversity of the solutions especially at the beginning of generation since multiple surrogates may build decision boundaries

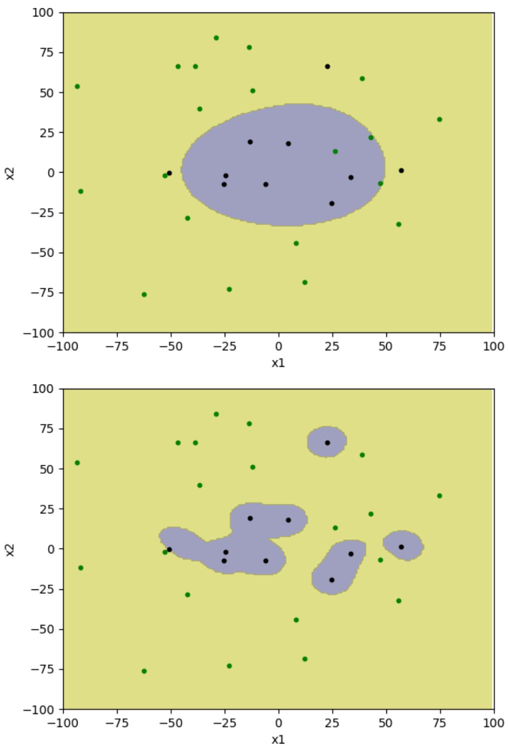

Fig. 2. Example of decision boundaries on a two dimensional classification task. The top and bottom figures demonstrate the decision boundaries obtained by SVM with low and high values of γ, respectively.

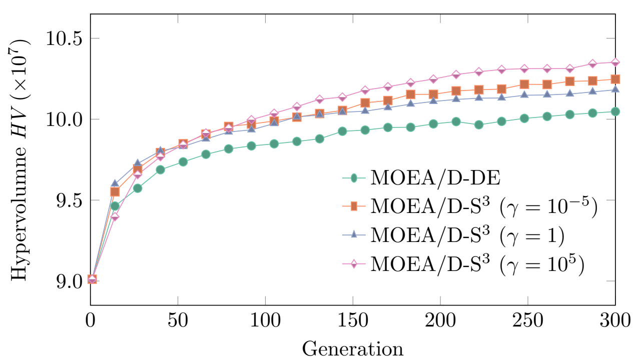

Fig. 3. Median values of HV scores of MOEA/D-S^3 with $\gamma = \{10^{-5}, 1.0, 10^5\}$ and that of MOEA/D-DE as a baseline (WFG1, $m = 8$).

capturing similar specific regions; although early solutions may dispersedly exist in the solution space but its diversity would be low in terms of the objective space. However, this drawback can be relaxed with the increase of generations, i.e., when each solution has been optimized for its subproblem, and thus multiple surrogates can capture their specific regions dispersedly existed in the solution space.

VII. Conclusion

This paper proposed a surrogate-assisted MOEA/D framework that employs multiple classifier-based surrogates. The proposed method, i.e., MOEA/D-S^3 is designed to construct reliable surrogate models under the limited number of samples and to improve a diversity of surrogate models under the samples biased in specific regions of the solution space. Accordingly, MOEA/D-S^3 employs the classifiers which can be given by solving a classification task having a lower problem complexity than fitness approximation; and each surrogate is adjusted to each subproblem defined in MOEA/D. We conducted the experiments on the benchmark problems, i.e., WFG1-9 problems. Experimental results showed that MOEA/D-S^3 significantly derived the better performance than MOEA/D-DE used in a basis of the proposed method. In addition, we further investigated a practical effect of the proposed parent-selection mechanism, which selects parents identified as good solutions; and our analysis showed the proposed selection mechanism can be an important option to improve the performance of MOEA/D-S^3.

As future works, we must compare MOEA/D-S^3 with the state-of-the-art of SAEAs, e.g. MOEA/D-EGO, CSEA, K-RVEA and we will conduct intensive experiments with different aspects; evaluation criteria e.g. IGD and the computational time, other benchmark problems as well as computationally-expensive real-world optimization problems. In addition, we can develop a self-adaptation method of the SVM parameter γ, as we revealed γ can be an important parameter to improve the performance of MOEA/D-S^3.

TABLE II
Median values of Hypervolume of MOEA/D-S^3 without our modified parent-selection; "+", "∼" and "−" represent that MOEA/D-S^3 without modification derives better, competitive, and worse performances than MOEA/D-S^3, respectively.

problem	m	2nd generation	10th generation	50th generation	300th generation
WFG1	4	8.67E+03 +	9.12E+03 −	9.70E+03 ∼	9.84E+03 ∼
	6	8.90E+05 +	9.35E+05 +	9.88E+05 ∼	9.99E+05 −
	8	9.04E+07 +	9.43E+07 −	1.00E+08 ∼	1.02E+08 ∼
WFG2	4	1.21E+04 +	1.34E+04 +	1.45E+04 +	1.45E+04 +
	6	1.25E+06 +	1.54E+06 ∼	1.73E+06 ∼	1.75E+06 ∼
	8	1.19E+08 −	1.75E+08 ∼	2.06E+08 −	2.10E+08 ∼
WFG3	4	1.10E+04 ∼	1.18E+04 ∼	1.25E+04 ∼	1.26E+04 −
	6	1.00E+06 −	1.11E+06 ∼	1.20E+06 −	1.23E+06 −
	8	8.27E+07 +	9.68E+07 +	1.06E+08 −	1.08E+08 −
WFG4	4	1.03E+04 −	1.26E+04 −	1.39E+04 −	1.41E+04 −
	6	5.92E+05 +	9.86E+05 +	1.47E+06 −	1.56E+06 −
	8	1.30E+07 +	3.14E+07 −	8.90E+07 −	1.10E+08 −
WFG5	4	1.10E+04 −	1.21E+04 ∼	1.35E+04 −	1.38E+04 −
	6	9.35E+05 +	1.23E+06 −	1.45E+06 −	1.51E+06 −
	8	4.83E+07 −	1.00E+08 −	1.46E+08 −	1.55E+08 −
WFG6	4	1.08E+04 ∼	1.25E+04 −	1.39E+04 ∼	1.40E+04 −
	6	7.97E+05 +	1.19E+06 −	1.56E+06 −	1.62E+06 −
	8	2.94E+07 +	7.34E+07 +	1.45E+08 +	1.58E+08 −
WFG7	4	1.11E+04 −	1.31E+04 −	1.41E+04 ∼	1.42E+04 −
	6	8.07E+05 +	1.25E+06 ∼	1.59E+06 −	1.64E+06 −
	8	2.14E+07 −	6.90E+07 −	1.44E+08 −	1.57E+08 −
WFG8	4	1.03E+04 ∼	1.19E+04 −	1.33E+04 ∼	1.35E+04 ∼
	6	6.91E+05 −	1.04E+06 −	1.33E+06 −	1.39E+06 −
	8	2.03E+07 −	4.85E+07 −	9.51E+07 −	1.04E+08 −
WFG9	4	1.11E+04 +	1.31E+04 ∼	1.40E+04 −	1.41E+04 −
	6	8.69E+05 +	1.38E+06 ∼	1.62E+06 −	1.65E+06 −
	8	4.49E+07 +	1.05E+08 −	1.78E+08 −	1.85E+08 −
+/−/∼		15/9/3	5/13/9	2/17/8	1/21/5

References

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

[2] 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 Transactions on Evolutionary Computation*, vol. 22, no. 1, pp. 129–142, 2016.

[3] Y. Jin, "A comprehensive survey of fitness approximation in evolutionary computation," *Soft computing*, vol. 9, no. 1, pp. 3–12, 2005.

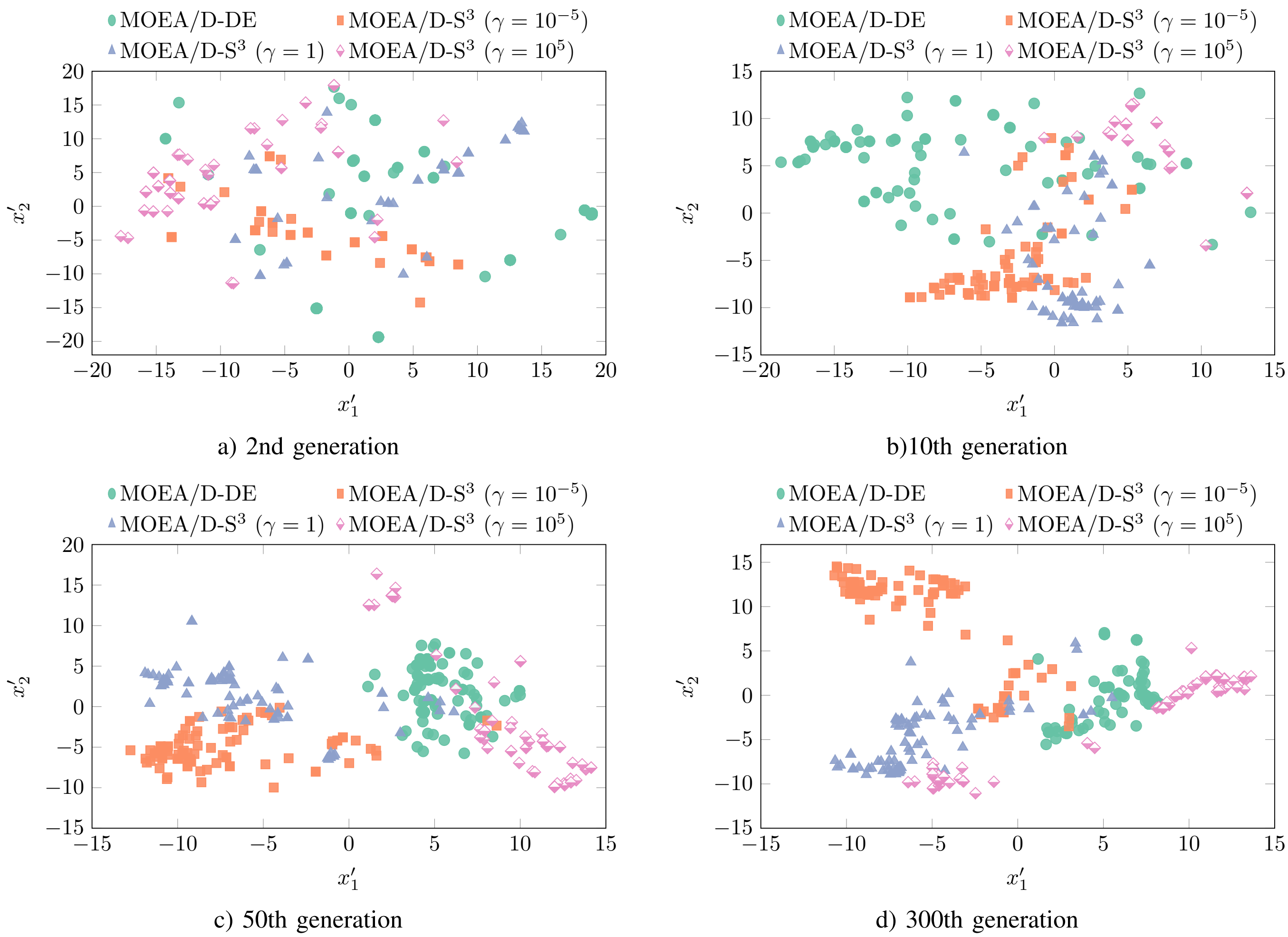

Fig. 4. Distributions of solutions obtained by MOEA/D-DE and MOEA/D-S^{3}with $\gamma = \{10^{-5}, 1.0, 10^5\}$ at 2nd, 10th, 50th and 300th generations (WFG1, $m = 8$). All solutions $\boldsymbol{x} \in \mathbb{R}^n$ are converted to two-dimensional vectors $\boldsymbol{x}' = \{x'_1, x'_2\}$ by t-SNE.

[4] A. Habib, H. K. Singh, T. Chugh, T. Ray, and K. Miettinen, "A multiple surrogate assisted decomposition based evolutionary algorithm for expensive multi/many-objective optimization," *IEEE Transactions on Evolutionary Computation*, 2019.

[5] A. Díaz-Manríquez, G. Toscano, J. H. Barron-Zambrano, and E. Tello-Leal, "A review of surrogate assisted multiobjective evolutionary algorithms," *Comp. intel. and neuroscience*, vol. 2016, p. 9420460, 2016.

[6] 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.

[7] 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, 2009.

[8] Q. Zhang and H. Li, "Moea/d: A multiobjective evolutionary algorithm based on decomposition," *IEEE Transactions on evolutionary computation*, vol. 11, no. 6, pp. 712–731, 2007.

[9] P. K. S. Nain and K. Deb, "A computationally effective multi-objective search and optimization technique using coarse-to-fine grain modeling," Kangal Report no. 2002005, 2002.

[10] S. Zapotecas Martínez and C. A. Coello Coello, "Moea/d assisted by rbf networks for expensive multi-objective optimization problems," in *Proceedings of the 15th annual conference on Genetic and evolutionary computation*. ACM, 2013, pp. 1405–1412.

[11] A. Rosales-Pérez, C. A. C. Coello, J. A. Gonzalez, C. A. Reyes-Garcia, and H. J. Escalante, "A hybrid surrogate-based approach for evolutionary multi-objective optimization," in *2013 IEEE Congress on Evolutionary Computation*. IEEE, 2013, pp. 2548–2555.

[12] J. Knowles, "Parego: a hybrid algorithm with on-line landscape approximation for expensive multiobjective optimization problems," *IEEE Trans. on Evolutionary Computation*, vol. 10, no. 1, pp. 50–66, 2006.

[13] C. K. Williams, "Prediction with gaussian processes: From linear regression to linear prediction and beyond," in *Learning in graphical models*. Springer, 1998, pp. 599–621.

[14] I. Loshchilov, M. Schoenauer, and M. Sebag, "A mono surrogate for multiobjective optimization," in *Genetic and evolutionary computation conference*. ACM, 2010, pp. 471–478.

[15] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: Nsga-ii," *IEEE transactions on evolutionary computation*, vol. 6, no. 2, pp. 182–197, 2002.

[16] H. Li and Q. Zhang, "Multiobjective optimization problems with complicated pareto sets, moea/d and nsga-ii," *IEEE transactions on evolutionary computation*, vol. 13, no. 2, pp. 284–302, 2009.

[17] B. E. Boser, I. M. Guyon, and V. N. Vapnik, "A training algorithm for optimal margin classifiers," in *Proceedings of the fifth annual workshop on Computational learning theory*. ACM, 1992, pp. 144–152.

[18] S. Huband, L. Barone, L. While, and P. Hingston, "A scalable multi-objective test problem toolkit," in *International Conference on Evolutionary Multi-Criterion Optimization*. Springer, 2005, pp. 280–295.

[19] E. Zitzler and L. Thiele, "Multiobjective optimization using evolutionary algorithms a comparative case study," in *International conference on parallel problem solving from nature*. Springer, 1998, pp. 292–301.

[20] F.-A. Fortin, F.-M. De Rainville, M.-A. Gardner, M. Parizeau, and C. Gagné, "DEAP: Evolutionary algorithms made easy," *Journal of Machine Learning Research*, vol. 13, pp. 2171–2175, jul 2012.

[21] L. v. d. Maaten and G. Hinton, "Visualizing data using t-sne," *Journal of machine learning research*, vol. 9, no. Nov, pp. 2579–2605, 2008.