# An Extended ISM for Globally Multimodal Function Optimization by Genetic Algorithms

Naoya Karatsu, Yuichi Nagata, Isao Ono, Shigenobu Kobayashi
Tokyo Institute of Technology
4529 Nagatsuta-cho, Midori-ku,
Yokohama-city, Kanagawa 226-8503, Japan
{karatsu, nagata}@fe.dis.titech.ac.jp, {kobayasi, isao}@dis.titech.ac.jp

***Abstract*—When attempting to optimize a function where exists several big-valley structures, conventional GAs often fail to find the global optimum. Innately Split Model (ISM) is a framework of GAs, which is designed to avoid this phenomenon called $UV$-$Phenomenon$. However, ISM doesn't care about previously-searched areas by the past populations. Thus, it is possible that populations of ISM waste evaluation cost for redundant searches reaching previously-found optima. In this paper, we introduce Extended ISM (EISM) that uses search information of past populations as $trap$ to suppress overlapping searches. To show performance of EISM, we apply it to some test functions, and analyze the behavior.**

## I. Introduction

In engineering, continuous function optimization is one of the important problems, and it is known that Real-Coded Genetic Algorithms (RCGAs) have a good performance on such problem. However, most of RCGAs are designed on the assumption that objective function has a $big$-$valley$[1] structure. Therefore, when attempting to optimize functions which have landscape called $globally\ multimodal$, where exist several big-valleys, RCGAs often fail to find the global optimum. Ikeda et al. called this phenomenon $UV$-$Phenomenon$ and classify causative structures into 3 classes by characteristics. Furthermore they explained how exploration fail and proposed Innately Split Model (ISM)[2] aiming to avoid UV-Phenomenon.

Populations in ISM called $groups$ are initialized within small area, and then they search independently. Moreover, converged groups are initialized to a random point. This prevents the groups from deceiving into local optimum. However, if the function has big-valleys which are larger than others, most of the groups search the large big-valleys. It means that the evaluation cost are wasted for the redundant searches. Although Ikeda et al. introduce the feature called $option$, it can be used in the limited situation and doesn't care about past searches. It occurs that the groups search previously-searched area and converge to previously-found optimum. In real world problems, cost for evaluate is very large and saving them is important task. Hence it is desirable to suppress the redundant searches of ISM and utilize resources efficiently.

This work introduces Extended-ISM (EISM) which uses distribution information of groups as $trap$ to represent previously-searched area. Each group saves their distribution information at some of their generations to $history$, and finally registers them to search space as traps. The traps are shared with all groups to find redundant searches. This action helps to utilize resource efficiency for finding the global optimum.

The following section describes the background of this work and motivation of us. In section 3, we propose EISM, which extends ISM, aiming to suppress redundant search. In section 4, to demonstrate performance, we apply EISM to test functions.

## II. Motivation and Background

In this section, we briefly review a background of our proposal and clarify motivation.

### A. UV-Structures

Ikeda et al. suggest that searches of GA fail to find the global optimum if objective functions have some big-valleys which have certain characteristics. The phenomenon are called UV-Phenomenon and such functions have landscapes called $UV$-$Structures$. In UV-Phenomenon Hypothesis, a big-valley which includes the global optimum is called the $opt$-$valley$, and a big-valley which includes the local optimum is called $local$-$valleys$. The UV-Structures are described as follows.

1. The average quality of the opt-valley is worse than of local-valleys.
2. The opt-valley is very narrow compared to local-valleys.
3. The complicity of the opt-valley is higher than local-valleys.

In these structures, a population of conventional GAs is deceived into a local-valley. We pick up UV-Structure class 2 and process of them represent to Figure 1.

### B. Innately Split Model

On the UV-Phenomenon Hypothesis, if a population of GA covers area that includes several big-valleys, it would fail to find the global optimum. To overcome this phenomenon, ISM searches by populations initialized within limited small area. They are called $groups$ and ISM performs as Algorithm 1.

In Algorithm 1, each $g_i$ is group of ISM which has $N$ individuals, and $g_i.\mu$ is the mean of the group $g_i$. Line 1 (and 7,9) is the most important for ISM. The groups are initialized around random point and it makes searches successful. A

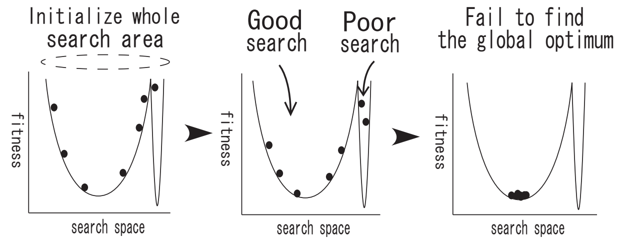


Fig. 1. Example of UV-Phenomenon: At first, a population of conventional GA is initialized whole search space (left). Then, in the course of the search, a difference of search speed between individuals which exist in the opt-valley and in the local-valley arise (middle). Finally all of individuals get pulled over to the local-valley (right).

**1** ISM

initialize $g_i (i = 1, \ldots, G)$ within small area
**while** not satisfy end condition **do**
    **for** $i := 1 : G$ **do**
        $g_i \leftarrow$ next generation of $g_i$
        **if** $g_i$ converges **then**
            save the optimum found by $g_i$
            initialize $g_i$ within small area
        **else if** any $j (j \neq i)$ satisfies $||g_i.\mu - g_j.\mu|| \leq r$ **then**
            initialize $g_i$ within small area
        **end if**
    **end for**
**end while**

group of ISM focuses on its search to the big-valley nearest from initialized position. Hence, initialize area should have similar or smaller size than the opt-valley.

The group which searches sufficiently is initialized at random position of search space. We call this $re\text{-}initialize$.

### *C. Motivation of our proposal*

If an objective function has landscape that belongs into UV-Structure class 2, most of groups are initialized in big-valley which has large volume, and then they converges to same optima. Furthermore, since ISM doesn't consider about previously-searched area, it occurs that the initialized group searches the area where is searched by past groups. This results that most of resources are wasted for the redundant searches.

One of the way to resolve this problem is to use tabu search[3]. However it finds the redundant search by the individual. Hence it is difficult to apply to RCGAs.

To suppress the redundant search, Ikeda et al. introduced a feature named $option$ that if the distance between two means of groups is shorter than threshold $r$, re-initialize one of them. However, the situation when the option can be used is limited because of following reason.

- **finding feasible value of $r$ is difficult :** Intuitively, $r$ requires shorter than the distance between the global optimum and nearest optimum from it. If $r$ is unfeasible, groups become to impossible to find certain optima.
- **option can be used for groups searching in parallel :** If ISM is not performed in parallel by the group, we cannot use the option. However, assuming we need high cost for evaluations, it is considered that it is efficient that using ISM in parallel by the individual.
- **option cannot deal with ill-scaled landscapes :** Because of option uses euclidean distance, option cannot deal with a function where exists ill-scaled landscape. If we apply ISM with option to such function, it could prevent groups from finding the global optimum.

To solve these problems, it is desirable that new method which is extending of option.

## III. PROPOSAL

In this section, we present Extended-ISM (EISM) which uses past search information to suppress redundant searches. Groups of EISM save their distribution information at several generation as $history$. Then, when the groups converge, they register history to search space. We call each registered information $trap$. Groups which search after them handle traps as previously-searched area. EISM performs as algorithm 2.

**2** EISM

1: initialize $g_i (i = 1, \ldots, G)$ within small area
2: **while** not satisfy end condition **do**
3:     **for** $i := 1 : G$ **do**
4:         $g_i \leftarrow$ next generation of $g_i$
5:         **if** $g_i$ converges **then**
6:             **register the history of** $g_i$ **to** $trap\_list$
7:             save the optimum found by $g_i$
8:             initialize $g_i$ within small area
9:         **else if** $\mathbf{capture\_checking}(g_i)$ **then**
10:             initialize $g_i$ within small area
11:         **end if**
12:         $\mathbf{add\_history}(g_i)$
13:     **end for**
14: **end while**

Most of the lines in Algorithm 2 are same with the lines in Algorithm 1. The trap_list is shared with all of the groups and it keeps traps. After performing the generation alternation, $g_i$ is compared with the traps (line 9). The function capture_checking($g_i$) returns TRUE if $g_i$ is determined to be re-initialized or FALSE if it isn't. If it is conclude that $g_i$ is captured by trap, $g_i$ is initialized without registering history (line 10). Then $g_i$ adds its distribution information to history as traps by the function add_history($g_i$). The function is depicted in Algorithm 3.

Figure 2 shows the search process of EISM.

### *A. Using a hyperellipsoid as a distribution information*

Assuming that each population of RCGAs follows normal distribution, we can represent its distribution by a hyperellipsoid using a covariance matrix as a coefficient matrix and a mean vector of it. If we use this hyperellipsoid as the trap, it can deal with ill-scaled landscape and dependence between variables.

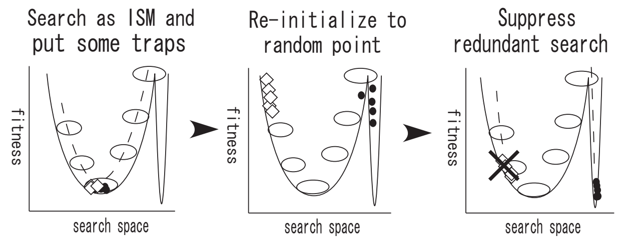


Fig. 2. Search process of EISM: The broken lines represents the search path ways of groups, and the mean of each group moves along the broken line. Group of ISM are initialized within small area and search independently, and then put the traps(left). The groups which converge are re-initialized(middle). A group which search similar area with past groups is captured by traps(right).

When there exists a trap using the mean $\mu$ and the coefficient matrix $\Sigma$, and a vector $\mathbf{x}$, we check whether $\mathbf{x}$ is included by trap or not using following functions:

$$D_m(\mathbf{x};\mu,\Sigma) = \sqrt{\frac{(\mathbf{x}-\mu)^T\Sigma^{-1}(\mathbf{x}-\mu)}{n}}. \tag{1}$$

If $C$ is $n \times n$ matrix as

$$C^T C = \Sigma^{-1}, \tag{2}$$

we can calculate mahalanobis distance as

$$\begin{aligned} \sqrt{(\mathbf{x}-\mu)^T\Sigma^{-1}(\mathbf{x}-\mu)} &= \sqrt{(\mathbf{x}-\mu)^T C^T C(\mathbf{x}-\mu)} && (3)\\ &= \sqrt{||C(\mathbf{x}-\mu)||^2}. && (4)\end{aligned}$$

In the function (4), $||C(\mathbf{x}-\mu)||^2 \sim \chi^2$ and expected value of it follows $n$. Thus, function (1) is normalized by $n$.

If $D_m(\mathbf{x};\mu,\Sigma) < \alpha$, we conclude that $\mathbf{x}$ exists in the trap.

### B. Putting traps on search space

Because of EISM aims suppress redundant searches for large local-valley in UV-Structure class 2, it is required to put a lot of traps in such valley. It is intuitive that the mean of a group moves frequently. On the other hand, it can be thought that making several traps at same place is not so important. From the view point, a group of EISM adds its distribution information to its history as follows:

**3** add_history($g$)

$\Sigma \leftarrow$ the covariance matrix of $g$
$\mu \leftarrow$ the mean vector of $g$
**if** $S_{pre}$ = nil **then**
  $S_{pre} \leftarrow \{\Sigma, \mu\}$
**else if** $D_m(\mu; S_{pre}.\mu, S_{pre}.\Sigma) > \alpha$ **then**
  $T \leftarrow \{\Sigma, \mu\}$
  $g.history\_list \leftarrow g.history\_list \cup T$
  $S_{pre} \leftarrow T$
**end if**

In Algorithm 3, $g$ is a group of EISM and $g.history\_list$ is a list which keeps distribution information of $g$. $S_{pre}$ and $T$ are variables that keep information of hyperellipsoid $\mu$, $\Sigma$ and we can get them as $S_{pre}.\mu$ and $S_{pre}.\Sigma$ respectively. $\alpha$ is a given parameter that determines expand ratio of traps. In this way, the group saves distribution information to history when the mean of the group is out of previous-saved history.

Then, the converged group registers the history as traps. The traps are put on as tracing search pathway of the group. When following groups encroach on any trap $T$, we define $the\ group\ is\ captured\ by\ T$. Then the captured groups are re-initialized in probability $P_{init}$.

As this way, suppressing redundant search effect of each trap is not so high. However, as we can observe in right of Figure 2, a group, which searches along past search path way, must be captured by a lot of traps, and re-initialized high probability. In contrast, if a group which searches local-valley makes some traps near by opt-valley. We call these traps $infeasible\ traps$. Groups searches the opt-valley is captured by a few of traps, and re-initialized low probability. This makes EISM performance robust.

### C. Capture checking

Each group is compared with all traps registered in search space to check captured or not. When more than half individuals of the group are in the trap, the group is captured by trap. Check of capture is performed as following function.

$$f(\mathbf{x};T) = \begin{cases} 1 & D_m(\mathbf{x};T.\mu,T.\Sigma) \le \alpha \\ 0 & otherwise \end{cases} \tag{5}$$

$$coverage(group;T) = \frac{\sum_{i=1}^{N} f(group.\mathbf{x}_i;T)}{N} \tag{6}$$

$group.\mathbf{x}_i$ is $i$th individual of group and $T.\Sigma, T.\mu$ are distribution information which are kept by a trap $T$. If there is $T* \in \mathcal{T} = \{\ t \mid t \in trap\_list,\ t \notin encounter\_list\}$ satisfies $coverage(group;T*) \ge 0.5$, group is captured by trap $T*$. Then the group is initialized in probability $P_{init}$ (the function capture_checking($g$) in Algorithm 2 returns TRUE) or add the trap to $encounter\_list$ (the function capture_checking($g$) in Algorithm 2 returns FALSE). In the later situation, we call that $the\ group\ avoids\ the\ trap\ T$, and the trap can't capture the group again. If remove this process and permit trap to capture the group again, a group which improves slow speed is captured many times by the same trap, and EISM lacks the robustness.

## IV. EXPERIMENTS

In this section, we perform experiments comparing ISM with EISM to present performance of EISM.

### A. Test functions

We use 4 functions for this experiment. We call 3 of the functions, Double-Sphere, Double-Rastrigin and Double-Rosenbrock, and we call them $Double\text{-}Valley\ functions$. Furthermore, another function is Fletcher and Powell function, and it is more complex than Double-Valley functions.

Double-Sphere is introduced in paper [4], and we adjust the position and the size of valleys for this experiment. Given real valued vector $\mathbf{x} \in R^n$ Double-Sphere is defined as follows:

$$\text{Double-Sphere}(\mathbf{x}) := min(f_{sph}(\mathbf{x}_s), f_{sph}(\mathbf{x}_l) + 1.0) \tag{7}$$

$$\mathbf{x}_l := \mathbf{x} - (2.56, \ldots, 2.56)^T \tag{8}$$

$$\mathbf{x}_s := 2(\mathbf{x} - (\text{-}2.56, \ldots, \text{-}2.56)^T) \tag{9}$$

$$f_{sph}(\mathbf{x}) := \sum_{i=1}^{n} x_i^2 \tag{10}$$

$$\mathbf{x} \in [-5.12, 5.12]^n \tag{11}$$

Double-Sphere has the local optimum at $(2.56, \ldots, 2.56)$ and the global optimum at $(\text{-}2.56, \ldots, \text{-}2.56)$ respectively. The volume of the local-valley is approximately $0.5^n$ times smaller than the opt-valley.

Double-Rastrigin is introduced in paper [4], and we adjust the position and the size of valleys for this experiment. Double-Rastrigin is defined as follows:

$$\text{Double-Rastrigin}(\mathbf{x}) := min(f_{ras}(\mathbf{x}_s), f_{ras}(\mathbf{x}_l) + 1.0) \tag{12}$$

$$\mathbf{x}_l := \mathbf{x} - (2.56, \ldots, 2.56)^T \tag{13}$$

$$\mathbf{x}_s := 2(\mathbf{x} - (\text{-}2.56, \ldots, \text{-}2.56)^T) \tag{14}$$

$$f_{ras}(\mathbf{x}) := f_{sph}(\mathbf{x}) - 10 \sum_{k=1}^{n} (1 - cos(2\pi(x_i - (\text{-}2.56)))) \tag{15}$$

$$\mathbf{x} \in [\text{-}5.12, 5.12]^n \tag{16}$$

Double-Rastrigin has the global optimum at $(\text{-}2.56, \ldots, \text{-}2.56)$ and a lot of local optima. The volume of the local-valley is approximately $0.5^n$ times smaller than the opt-valley.

Double-Rosenbrock is introduced in paper [5], and we adjust the position and the size of valleys for this experiment. Double-Rosenbrock is defined as follows:

$$\text{Double-Rosenbrock}(\mathbf{x}) := min(f_{ros}(\mathbf{x}_s), f_{ros}(\mathbf{x}_l) + 0.1) \tag{17}$$

$$\mathbf{x}_l := \mathbf{x} - (0.5, \ldots, 0.5)^T \tag{18}$$

$$\mathbf{x}_s := -2(\mathbf{x} - (\text{-}1.0, \ldots, \text{-}1.0)^T) \tag{19}$$

$$f_{ros}(\mathbf{x}) := \sum_{i=1}^{n} (100(x_1 - x_i^2)^2 + (1 - x_i)^2) \tag{20}$$

$$\mathbf{x} \in [\text{-}2.048, 2.048]^n \tag{21}$$

Double-Rastrigin has the local optimum at $(\text{-}1.5, \ldots, \text{-}1.5)$ and the global optimum at $(1.5, \ldots, 1.5)$ respectively. Groups searching the opt-valley once go to $(0.5, \ldots, 0.5)$ and groups searching the opt-valley once go to $(\text{-}1.0, \ldots, \text{-}1.0)$, then move along ridge structure.

Fletcher and Powell function is introduced in paper [6] and it has complicated landscape than Double-Valley functions. Fletcher and Powell function is defined as follows:

$$F(\mathbf{x}) := \sum_{i=1}^{n} (A_i - B_i)^2 \quad for \quad n = 12 \tag{22}$$

$$A_i := \sum_{j=1}^{n} (a_{ij} \sin\alpha_j + b_{ij} \cos\alpha_j) \tag{23}$$

$$B_i(\mathbf{x}) := \sum_{j=1}^{n} (a_{ij} \sin x_j + b_{ij} \cos x_j) \tag{24}$$

$a_{ij}$ and $b_{ij}$ are integer random numbers in the range $[\text{-}100, 100]$, and $\alpha_i$ are random numbers in the range $[\text{-}\pi, \pi]$. We use these values as introduced in paper[6]. On dimension 12, it is known that this function has 4 global-optima and some local-optima. TABLE I shows the names corresponding to all of the global-optima and top 3 local-optima of fitness, size and coordinate of them. Here, let size of optimum A is 1.0 and size of others are volume in relation of optimum A. We approximately measure the volume using 40,000 uniformly point and search the correspondences between them and the optima by conjugate gradient method.

TABLE I
PRIMARY OPTIMA OF FLETCHER AND POWELL FUNCTION : THE SIZE OF EACH OPTIMA IN RELATION TO OPTIMUM A, AND FITNESS AND COORDINATE OF THEM

| optimum | fitness | size | coordinate |
|---|---|---|---|
| A | 0 | 1.00 | $(0.44, 0.55, \ldots)$ |
| B | 0 | 1.47 | $(0.34, 0.47, \ldots)$ |
| C | 0 | 1.63 | $(1.22, 0.38, \ldots)$ |
| D | 0 | 1.52 | $(0.41, 0.39, \ldots)$ |
| a | 0.813 | 3.96 | $(0.35, \text{-}1.59, \ldots)$ |
| b | 10.62 | 2.83 | $(1.48, 0.70, \ldots)$ |
| c | 13.28 | 10.4 | $(\text{-}0.15, \text{-}1.29, \ldots)$ |

TABLE I indicates that size of optimum c is most large than others. Thus we can predict that the big-valley which includes optimum c is causes UV-Phenomenon and the landscape is categorized to UV-Structure class 2.

### B. Operators of GA

We use $REX^{star}(U, n+1)$[7] as crossover operator and $JGG$[8] as selection.

Given dimension $n$, $REX^{star}(U, n{+}1)$ selects $n{+}1$ parents and drives a globally gradient orientation, that represent the gradient of underlying structure. Then, let $x_b$ is the point which shifted toward a globally gradient orientation from mean of parents. The offsprings are produced around the $x_b$. From this way, because of population move to more feasible area, it prevents population from primary convergence. On the other hand, JGG selects the parents from population and puts them to decent individuals of offsprings. Since population of ISM, which is initialized in limited area, almost can't cover the optima and need to move toward better area, we verified that this combination was efficient on preliminary experiments.

### C. Settings

On Double-Valley functions, dimension of function $n = 10$ and setting of operators are determined by preliminary experiments as follows:

- On Double-Sphere, number of individuals in group $N$ is $3 \times n = 30$, number of children is $2 \times n = 20$, step size parameter of $REX^{star}$ is 8.

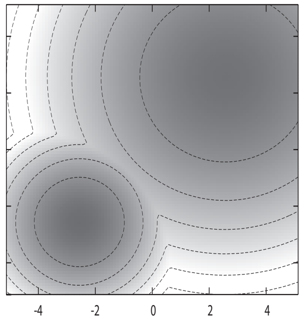

Fig. 3. 2 dimensional Double-Sphere

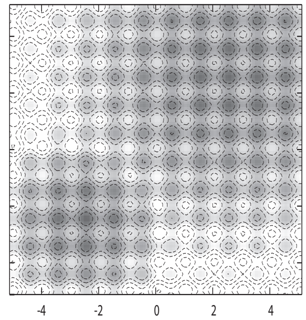

Fig. 4. 2 dimensional Double-Rastrigin

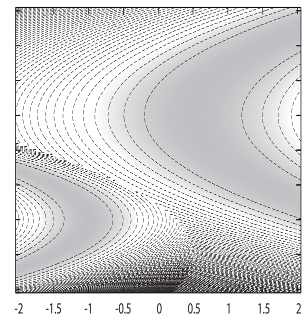

Fig. 5. 2 dimensional Double-Rosenbrock

- On Double-Rastrigin, number of individuals in group $N$ is $20 \times n = 200$, number of children is $3 \times n = 30$, step size parameter of $REX^{star}$ is 2.
- On Double-Rosenbrock, number of individuals in group $N$ is $5 \times n = 50$, number of children is $3 \times n = 30$, step size parameter of $REX^{star}$ is 4.

During the experiments, fitness of children that exist out side of the boundary region is infinity.

On Fletcher and Powell function, dimension of function $n = 12$ and setting of operators are determined by preliminary experiments as follows:

- The number of individuals in group $N$ is $8 \times n = 96$, number of children is $8 \times n = 96$, step size parameter of $REX^{star}$ is 4.

We handle search space as torus in the range $[\pi, -\pi]$.

Thus on all of the test functions, one side of initialized area is 0.3 times smaller than one side of boundary space on ISM and EISM commonly. It is sufficiently small to avoid the UV-Phenomenon, and we confirmed by the preliminary experiment that this setting is most efficient for ISM. In this experiment, the number of groups $G$ is 1. This is determined from the view point of efficiency as we suggest in section II. If ISM is parallelized by group, we can use EISM with option of ISM, however we don't deal it in this paper.

The parameters of EISM $\alpha$, which determines expand ratio of traps, is 1.5, $P_{init}$, which determines re-initialize probability, is 0.3, 0.5, 0.7, 0.9 for all functions. Each group is initialized when fitness gain of it is less than $10^{-7}$ for fifteen generations, because of the group converges. Each trial is until find the global optimum on Double-Valley functions, and until find the all of primary optima on Fletcher and Powell function. We examine 300 trials for each test function.

### *D. Results*

TABLE II shows avg. and s.d. on all of the test function.

**Double-Sphere**: EISM finds global optima faster than ISM, and the number of evaluation is approximately 25%. Furthermore the evaluation cost and the standard deviation are most low at $P_{init} = 0.9$.

**Double-Rastrigin**: EISM finds global optima faster than ISM, and the number of evaluation is approximately 30%. One characteristic point is that when $P_{init} = 0.9$, the number of evaluation and the standard deviation are larger than others.

**Double-Rosenbrock**: It indicates that the effect suppressing redundant search of EISM is better than that on other functions. It spends approximately 10% evaluations. On the other hand, it is observed that the standard deviation at $P_{init} = 0.9$ in this function becomes very high value.

**Fletcher and Powell function**: EISM finds all of primary optima faster than ISM, and the number of evaluations is approximately 25%. Thus we can say that standard deviation of them are low values against number of evaluations of them. However it is observed that standard deviation at $P_{init} = 0.9$ in this function becomes high.

TABLE II
RESULTS OF EXPERIMENTS: THE NUMBER OF EVALUATIONS AND STANDARD DEVIATION OF IT FOR ISM AND EISM.

Double-Sphere

| method | ISM | EISM | | | |
|---|---|---|---|---|---|
| $P_{init}$ | - | 0.3 | 0.5 | 0.7 | 0.9 |
| avg. ($\times 10^4$) | 33.3 | 9.51 | 9.58 | 8.83 | **7.94** |
| s.d. ($\times 10^4$) | 33.2 | 6.76 | 7.88 | 6.95 | 6.48 |

Double-Rastrigin

| method | ISM | EISM | | | |
|---|---|---|---|---|---|
| $P_{init}$ | - | 0.3 | 0.5 | 0.7 | 0.9 |
| avg. ($\times 10^5$) | 22.3 | 9.39 | 7.95 | **7.37** | 7.87 |
| s.d. ($\times 10^5$) | 22.3 | 6.62 | 5.68 | 5.19 | 12.8 |

Double-Rosenbrock

| method | ISM | EISM | | | |
|---|---|---|---|---|---|
| $P_{init}$ | - | 0.3 | 0.5 | 0.7 | 0.9 |
| avg. ($\times 10^6$) | 11.9 | 1.10 | **0.92** | 0.96 | 1.01 |
| s.d. ($\times 10^6$) | 11.5 | 0.83 | 0.66 | 0.79 | 1.17 |

Fletcher and Powell function

| method | ISM | EISM | | | |
|---|---|---|---|---|---|
| $P_{init}$ | - | 0.3 | 0.5 | 0.7 | 0.9 |
| avg. ($\times 10^6$) | 7.96 | **1.42** | 1.60 | 1.90 | 2.01 |
| s.d. ($\times 10^6$) | 4.96 | 0.63 | 0.91 | 1.52 | 1.74 |

TABLE III
THE NUMBER OF REDUNDANT SEARCHES ON DOUBLE-VALLEY FUNCTIONS.

| | Double-Sphere | Double-Rastrigin | Double-Rosenbrock |
|---|---|---|---|
| ISM | 48.6 | 35.9 | 178 |
| EISM | 2.00 | 4.39 | 1.78 |

TABLE IV
THE NUMBER OF REDUNDANT SEARCHES ON FLETCHER AND POWELL FUNCTION BY ISM AND EISM.

| | A | B | C | D | a | b | c |
|---|---|---|---|---|---|---|---|
| ISM | 2.03 | 2.64 | 1.64 | 0.59 | 5.09 | 2.60 | 21.7 |
| EISM | 0.29 | 0.46 | 0.16 | 0.16 | 0.68 | 0.55 | 1.30 |

### *E. Discussion*

*1) **Suppressing redundant search by traps**:* From the result of experiments, it is observed that EISM makes searches more efficient. To examine effect to suppress redundant searches of EISM, we count groups which converge to the local optimum on Double-Valley functions. Table III shows the results of 300 trial average. We can observe that many groups found local-optima by ISM, and we can say that they are redundant searches. On the other hand, few groups of EISM found the local-optima. These results indicate that EISM can suppress search for previously-find optima with high probability. Especially, effect on Double-Rosenbrock is very well.

On Double-Rosenbrock, it is known that groups which converge to same optimum follow same pathway. In such situation, a group of EISM, which converge first to the optimum, put several traps on the pathway, and following groups are captured by the traps with high probability. It is known that function where exists a ridge structure needs high evaluation cost and the waste of redundant search is prominence. While our proposal is particularly efficient to such landscape, we can say that it is a strong point of our proposal.

TABLE IV shows number of groups which converge to each optimum of Fletcher and Powell function. By ISM, it can be observed that optimum c is found more than 22 times. It indicates that most of groups search to local-valley where exists optimum c. On the other hand, EISM finds optimum c approximately 2 times and find others approximately once.

Furthermore we examine the big-valleys where is searched by groups which are re-initialized by traps on Fletcher and Powell function. It is examined by the way that the groups determined to be re-initialized continue to search until converge, and we count evaluation costs which spent for the search interrupted by the trap. TABLE V shows the number of groups which searched on each big-valley and suppressed generations of them on Fletcher and Powell function. The groups that searched for optimum c is most re-initialized by traps. Because of optimum c is largest size of all of the primary optima, the result is suitable for our motivation.

TABLE V
THE NUMBER OF GROUPS RE-INITIALIZED BY TRAPS AND RATIO OF SUPPRESSED GENERATIONS.

| | A | B | C | D | a | b | c |
|---|---|---|---|---|---|---|---|
| re-init | 1.61 | 1.95 | 0.37 | 2.00 | 4.08 | 2.30 | 20.7 |
| suppress | 84% | 80% | 89% | 79% | 89% | 86% | 96% |

*2) **Effect of probabilistic re-initialize**:* Now focus on the results with $P_{init} = 0.9$ on TABLE II. We can find that number of evaluations and standard deviation is larger than result with $P_{init} = 0.7$ on Double-Rastrigin and Double-Rosenbrock. It implies that there are something interrupt the searches to find the global optima. We can estimate that infeasible traps cause this result, though the Double-Valley functions have simple landscape. Moreover It is observed on Fletcher and Powell function in TABLE II that the standard deviation of EISM of with a high $P_{init}$ is higher than that of others. If groups which captured by infeasible traps are re-initialized with high probability, it is difficult to find the optimum. Hence it is important for EISM to let $P_{init}$ low value, and we recommend to be $P_{init} = 0.5$. In this way, EISM can deal more complicated problems.

Performing EISM with the low $P_{init}$ makes the effect of suppressing redundant searches lower. There is trade-off between the effect and the robustness. It is our future works that to improve the effect of EISM with left the robustness.

## V. CONCLUSIONS

In this paper, we noted the waste of resources of ISM. To address these concerns, we proposed EISM which uses information of areas where were previously-searched by past groups, and suppresses redundant searches. We applied it to test functions which have UV-Structure class 2 and Fletcher and Powell function that has complex landscape than them, and showed that EISM can find objective optimum with the smaller number of evaluations than ISM and EISM have high performance especially on functions that have a ridge structure. Furthermore we can appeal that EISM can be applied to various functions without adjusting parameters.

## REFERENCES


[1] K. D. Boese, "Cost versus distance in the traveling salesman problem," in *Cost Versus Distance In the Traveling Salesman Problem*. ACM, 1995, pp. 254 – 269.

[2] K. Ikeda and S. Kobayashi, "GA Based on the UV-structure Hypothesis and Its Application to JSP," *Lecture Notes In Computer Science*, vol. 1917, no. 3, pp. 273–282, 2000.

[3] Y. Hanada, T. Hiroyasu, and M. Miki, "Tabu: local search mechanism for mega process genetic algorithm," in *2004 IEEE Conference on Cybernetics and Intelligent Systems*, vol. 1, no. 4, 2004.

[4] M. Lunacek, D. Whitley, and A. Sutton, "The impact of global structure on search," in *Preceedings of the 10th International Conference on Parallel Problem Solving from Nature PPSM X*, 2008, pp. 498 – 507.

[5] S. Kinoshita, J. Sakuma, I. Ono, and S. Kobayashi, "Introduce and evaluate new framework for real-coded ga considering uv-structure," in *The 22nd Annual Conference of The Japanese Society for Artificial Intelligence*, 2008.(In Japanese)

[6] M. Bäck, *Evolutionary algorithms in theory and practice*. Oxford university press, 1996.

[7] S. Kobayashi, "The frontiers of real-coded genetic algorithms," *Journal of Japanese Society for Artificial Intelligence*, vol. 24, no. 1, pp. 147 – 162, 2009.(In Japanese)

[8] Y. Akimoto, R. Hasada, J. Sakuma, I. Ono, and S. Kobayashi, "Generation Alternation Model for Real-Coded GA Using Multi-parent Proposal and Evaluation of Just Generation Gap(JGG)." in *In Proceedings of the 19th SICE Symposium on Decentralized Antonomous Systems*, 2007, pp. 341–346.(In Japanese)