

# Latent Variable Crossover for $k$ -tablet Structures and its Application to Lens Design Problems

Jun Sakuma  
Tokyo Institute of Technology  
4259, G5-21, Nagatsuta-cho  
Midori-ku, Yokohama-city, Kanagawa, Japan  
jun@fe.dis.titech.ac.jp

Shigenobu Kobayashi  
Tokyo Institute of Technology  
4259, G5-21, Nagatsuta-cho  
Midori-ku, Yokohama-city, Kanagawa, Japan  
kobayasi@dis.titech.ac.jp

## ABSTRACT

This paper presents the Real-coded Genetic Algorithms for high-dimensional ill-scaled structures, what is called, the  $k$ -tablet structure. The  $k$ -tablet structure is the landscape that the scale of the fitness function is different between a  $k$ -dimensional subspace and the orthogonal  $(n-k)$ -dimensional subspace. The search speed of traditional GAs degrades when a high dimensional  $k$ -tablet structure is included in the landscape of the fitness function.

In this structure, offspring generated by crossovers are likely to spread wider region than the region where the parental population covers and this causes the stagnation of the search. To resolve this problem, we propose a new crossover LUNDX- $m$  using only  $m$ -dimensional latent variables. The effectiveness of the proposal method is tested with several benchmark functions including  $k$ -tablet structures and we show that our proposed method performs better than traditional crossovers especially when the dimensionality  $n$  is higher than 100.

As an example of a  $k$ -tablet structure in real world applications, we show that the lens design problem has a kind of  $k$ -tablet structures and that our proposed method also performs better than conventional crossovers in this problem.

## Categories and Subject Descriptors

F.2.1 [Theory of Computation]: Analysis of Algorithms and Problem Complexity numerical algorithms and problems

## General Terms

Algorithms

## Keywords

Genetic Algorithm, Optimization, landscape,  $k$ -tablet, Crossover, Real coded, Real parameter, Latent variable

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

GECCO'05, June 25–29, 2005, Washington, DC, USA.  
Copyright 2005 ACM 1-59593-010-8/05/0006 ...\$5.00.

## 1. INTRODUCTION

Function optimization in the continuous search space is one of the most important problems because it often has to be solved in real-world applications. In function optimization, high-dimensionality, strong epistasis (dependency between parameters) and numerous local optima make the optimization difficult. Real-coded Genetic Algorithms (RC-GAs) are expected to be a powerful optimizer that can solve problems with these difficulties.

In this paper, we focus on ill-scaled and epistasis functions. For crossovers invariant to the coordinate system, ill-scaledness and epistasis can be treated equivalently, therefore, we generalize these properties as  $k$ -tablet structures. In optimization of  $n$  dimensional functions, when the sensitivity to the cost is largely different between  $k$  directions and the other orthogonal  $(n-k)$  directions, we call this structure of the landscape a  $k$ -tablet structure.

$k$ -tablet structures often arise in real world applications. The lens design problem is known as a difficult optimization problem with numerous local optima and complicated constraints[4]. We show that this problem is a typical example of a cost function with  $k$ -tablet structures. When a function has a  $k$ -tablet structure, the convergence speed seriously degrades. Although crossover UNDX- $m$ [3] and SPX[8] are known as crossovers which cannot be effected by ill-scaledness, the convergence speed of RCGAs with these crossovers become considerably slower as the dimensionality of the function becomes higher in actual.

The purpose of this paper is to study about the reason of inefficiency of conventional crossovers in high dimensional  $k$ -tablet structures. Then, we propose a new scalable RCGA to dimensionality. The performance of proposed method is verified by experiments using artificial benchmark functions. Then, we study about  $k$ -tablet structures in the lens design problem. We show that our proposed method solves this problem more efficiently than conventional RCGAs.

In section 2,  $k$ -tablet structures are defined. Section 3 clarifies problems of conventional methods in  $k$ -tablet structures. In section 4, a new RCGA for  $k$ -tablet structures is proposed. Section 5 is experiments. In section 6,  $k$ -tablet structures in lens design problems are analyzed and the proposed method is applied. Section 7 concludes this paper.

## 2. K-TABLET STRUCTURES

### 2.1 Ill-scaledness and Epistasis

Ill-scaledness means that the scales between variables are

largely different each other. From the viewpoint of the cost function, this means that the sensitivity to the cost is different dependent on the variable. Epistasis means dependency between variables. In epistasis functions, the cost cannot be improved when one of dependent variables is changed but improved when all of dependent variables are appropriately changed together. Especially in high dimensionality, the dependency is likely to be complicated.

Ill-scaled functions and epistasis functions can be regarded as equivalent structures when the coordinate system is taken in local region properly. If the optimization algorithm works independently on the given coordinate system, that is, the algorithm is coordinate system invariant, then both functions can be treated equivalently.

As a benchmark of epistasis functions, Rosenblock function is often used. In this function, the landscape is a gentle descent in a direction and in the other  $n - 1$ -dimensional directions the landscape is a steep ascent. This kind of structure is called ‘‘ridge structure’’[5]. Epistasis in Rosenblock function cannot be resolved just by taking an appropriate coordinate system because the shape of the ridge is a curve, however, on a local region, it can be regarded as resolved approximately by a linear transformation.

In high dimensional problems, the epistasis or ill-scaled structures do not only mean ridge structures but more generalized structures which have a gentle descent in  $k$ -dimensional directions and in the other  $n - k$ -dimensional directions the landscape is a steep ascent. The  $k$ -tablet structure is defined as a generalized epistasis or ill-scaled structure.

## 2.2 Definition of $k$ -tablet Structure

We consider minimizing a nonlinear cost function  $f(\mathbf{x})$  that is a mapping from a search space  $S \subset \mathbf{R}^n$  to  $\mathbf{R}$ , where the variable is  $\mathbf{x} \in \mathbf{R}^n$ . Suppose that  $f(\mathbf{x})$  is locally approximated as a quadratic form  $f(\mathbf{y}) = (\mathbf{A}\mathbf{y})^T \mathbf{A}\mathbf{y} = \mathbf{y}^T \mathbf{H}\mathbf{y}$  around  $\mathbf{x}_0$ , where  $\mathbf{y} = \mathbf{x} - \mathbf{x}_0$ ,  $\mathbf{A}$  is a full-rank matrix and  $\mathbf{H}$  is a symmetric and positive-definite Hessian matrix. When we choose an arbitrary orthogonal matrix  $\mathbf{P}$ , then  $f(\mathbf{y})$  is written in the form  $f(\mathbf{y}) = \alpha_1 z_1^2 + \dots + \alpha_n z_n^2$ , where  $\mathbf{y} = \mathbf{P}\mathbf{z}$ ,  $\alpha_i (i = 1, \dots, n)$  are eigenvalues of  $\mathbf{H}$  and eigenvalues is ordered as  $\alpha_1 \geq \alpha_2 \geq \dots \geq \alpha_n$ .

When  $k$  eigenvalues  $\alpha_1, \dots, \alpha_k$  is much larger than the other eigenvalues  $\alpha_{k+1}, \dots, \alpha_n$ , then we call  $f(\mathbf{x})$  a  $k$ -tablet structure around  $\mathbf{x}_0$ . Similarly, a Gaussian distribution with the covariance matrix having the same property is called a  $k$ -tablet Gaussian distribution. Data set  $G = \{\mathbf{x}_i | \mathbf{x}_i \in \mathbf{R}^n, i = 1, \dots, N\}$  drawn from a  $k$ -tablet Gaussian distribution can be approximately represented by a linear combination of  $k (< n)$  large eigenvectors  $(\mathbf{e}_1, \dots, \mathbf{e}_k)$  as follows:

$$\mathbf{x}_i = \sum_{j=1}^k \mathbf{e}_j t_{i,j} + \epsilon, \quad (1)$$

where  $\mathbf{t}_i \in \mathbf{R}^k$ ,  $t_{i,j}$  is a  $j$ -th element of  $\mathbf{t}_i$  and  $\epsilon$  is a  $n$  dimensional random value following a Gaussian distribution with a sufficiently small variance, that is,  $\epsilon \sim N(\mathbf{0}, \sigma^2 \mathbf{I})$ . Then  $\mathbf{e}_1, \dots, \mathbf{e}_k$  are called latent variables[7]. The data model described as eq.( 1) is called a  $k$ -dimensional latent variable model( $k$ -LVM). The  $k$ -dimensional subspace that latent variables span is called  $k$ -tablet space. Fig. 1 shows a conceptual illustration of an example of 3 dimensional contours of  $k$ -tablet structures and  $k$ -LVMs.

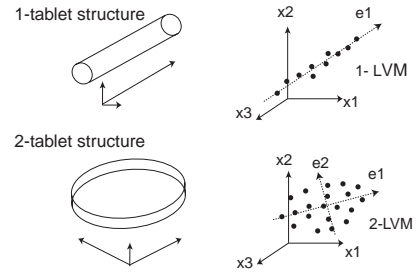


Figure 1: Conceptual illustrations of contours of  $k$ -tablet structures and  $k$ -LVMs

## 2.3 Situations $k$ -tablet Structures Arise

A typical situation where  $k$ -tablet structures arise is that the cost function has variables which are insensitive to the cost. In this situation, the gradient is quite small or zero in the direction of insensitive variables. Therefore, cost function forms a  $k$ -tablet structure. Redundancy or symmetric property also induces  $k$ -tablet structures. For example, in the weight optimization of neural networks, equivalent neural networks can be obtained even if the weight variables have different values because of the redundancy. These situations can be avoided if the cost function is deigned carefully but when the dimensionality is extremely high, removing these insensitivity or redundancy is difficult.

The difference of physical dimensions also induces  $k$ -tablet structures. When the variables have the same physical dimension but the scale is different (for example, meter and millimeter), a linear transformation of the domain resolves the ill-scaledness. However, when the physical dimensions are different, transformations to resolve the ill-scaledness are not known. Moreover, when there exist complicated interdependencies between them, finding an appropriate transformation that resolves the epistasis is difficult.

## 3. PROBLEMS OF CONVENTIONAL METHODS

### 3.1 Guideline of Designing Crossovers

Given the assumption that the population size is sufficiently large, Kita et al. have proposed a hypothesis on the behavior of the population of a GA as follows[3]: information gathered by the GA run is represented by this population, and it suggests that the region where the population resides is a promising region for further search.

Then, as a guideline for sampling new points in the search space, it has been suggested that offspring generated by a crossover should be approximately distributed over the similar region where the current population covers. If the offspring generated by a crossover distribute narrower than the parents, it may let the optimum escape. On the contrary, if the offspring generated by a crossover distribute wider, it wastes computation time in searching hopeless region. Hence, sampling new points in the region where the parents reside will be an appropriate choice.

In order to satisfy the condition described above, the preservation of the statistics, such as the mean vector or the covariance matrix, has been proposed. Please notice that the preservation of the covariance matrix is important from the viewpoint of optimization of epistasis functions.

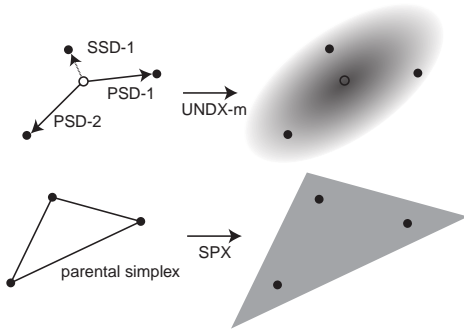


Figure 2: UNDX- $m$  and SPX

If the cost function has epistasis and the parental population reflects the correlated valley well, the distribution of the population is expected to have the correlation. In order to give the same correlation to offspring, crossovers must preserve not only the mean vector but also the covariance matrix.

### 3.2 Hopeful Crossovers in $k$ -tablet Structures

In this section, we discuss about a hopeful sampling strategy for a crossover on  $k$ -tablet structures. When the cost function has a  $k$ -tablet structure, the parental population hopefully forms  $k$ -LVM reflecting the  $k$ -tablet structure. Therefore, intensive sampling in the  $k$ -tablet space of the parental population is expected to contribute to effective search. On the other hand, if the crossover generates offspring in the region far from the parental population, the computation time is wasted on the search in the unhopeful region. Hence, when the parental population forms an appropriate  $k$ -LVM, it is desired that the offspring generated by a crossover also should form the same  $k$ -LVM. In this paper, we call this property the *preservation of  $k$ -LVM*.

In next section, we analyze the efficiency of conventional crossovers which preserve the covariance of the population and show that they do not necessarily perform better in  $k$ -tablet structures.

### 3.3 UNDX- $m$ in $k$ -tablet Structures

Multiple-parental extension of Unimodal Normal Distribution Crossover (UNDX- $m$ ) [3] generates offspring following the normal distribution spanned by  $m+2$  parents  $x^1, \dots, x^{m+2}$ . Let the center of parental vectors  $x^1, \dots, x^{m+1}$  be  $\mathbf{p}$ , the difference vector of  $x^i$  and  $\mathbf{p}$  be  $\mathbf{d}^i = x^i - \mathbf{p}$ . Let  $D$  be the length of the component of  $\mathbf{d}^{m+2} = x^{m+2} - \mathbf{p}$ . Let  $e^1, \dots, e^{n-m}$  be an orthogonal bases of the subspace orthogonal to the subspace spanned by  $\mathbf{d}^1, \dots, \mathbf{d}^m$ . Then, offspring vector  $x^c$  is generated as follows:

$$\mathbf{x}^c = \mathbf{p} + \sum_{i=1}^m w_i \mathbf{d}^i + \sum_{i=1}^{n-m} v_i D e^i \quad (2)$$

, where  $w_i, v_i$  is a random number following a normal distribution  $N(0, \sigma_\xi^2), N(0, \sigma_\eta^2)$  respectively.  $\sigma_\xi = \alpha/\sqrt{m}, \sigma_\eta = \frac{1}{\sqrt{n-m}} \frac{\sqrt{m+1}}{\sqrt{m+2}} \frac{3}{2} \beta, \alpha = 1, \beta = 0.35$  is recommended in terms of the preservation of statistics in [3]. Fig. 2 shows the distribution of offspring generated by UNDX- $m$ .

In eq. (2), the second and third term of RHS are called primary search directions(PSDs) and secondary search di-

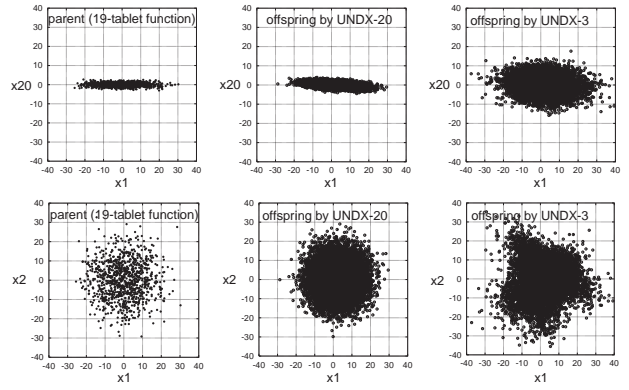


Figure 3: Left: Parental population on 20-dimensional 19-tablet structures, center: offspring generated by UNDX-20, right: offspring generated by UNDX-3. The vectors are projected on to  $x_1 - x_{20}$  space (top) and  $x_1 - x_2$  space (bottom).

rections(SSDs) respectively. PSDs satisfy the preservation of statistics but SSDs do not. Therefore, the preservation of  $k$ -LVM is not completely satisfied by UNDX- $m$ .

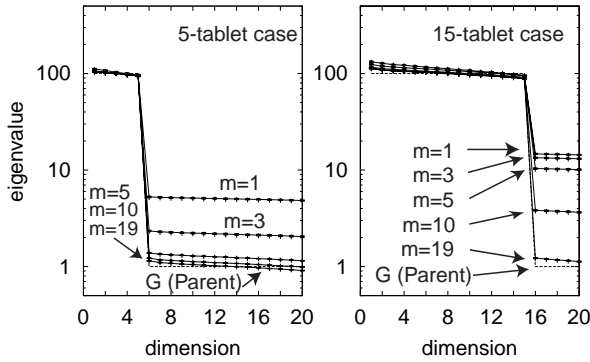
Fig. 3 shows offspring generated by UNDX-3 and UNDX-20 when the parental population is a 20-dimensional 19-tablet normal distribution. The figure suggests that offspring generated by UNDX-20 preserves the parental  $k$ -LVM but UNDX-3 does not. Figure 4 shows another numerical examples. Let  $G$  be a parental population that is drawn from 5-tablet and 15-tablet normal distribution, where the mean vectors are at the origin  $(0, \dots, 0)$ , the covariance matrices

are  $\text{diag}(\overbrace{100, \dots, 100}^k, \overbrace{1, \dots, 1}^{n-k})$  and the dimension  $n = 20$ . Let  $O$  be offspring generated by UNDX- $m$ , where the parental vectors are chosen randomly from  $G$ .

The form of the parental population or offspring can be estimated by applying primary component analysis (PCA) to the vectors and comparing eigenvalues obtained as a result of the PCA. When  $k$  eigenvalues are much larger than the other  $n - k$  eigenvalues, then the population forms  $k$ -LVM.

Figure 4 shows the eigenvalues of each component of  $G$  and  $O$  which are obtained by PCA and ordered by the size. In 5-tablet case, the eigenvalues of  $O$  are similar to those of  $G$  at  $m \geq k (m = 5, 10, 19)$ , but the eigenvalues of  $O$  are biased to be much larger than those of  $G$  at  $m < k (m = 1, 3)$ . Similarly, in 15-tablet case, eigenvalues of  $O$  are biased to be larger than those of  $G$  at  $m < k (m = 1, \dots, 10)$ . These results suggest that UNDX- $m$  does not preserve  $k$ -LVM when the  $k$  of  $k$ -tablet is larger than the number of parents  $m$ .

Figure 5 explains why the eigenvalues of offspring generated by UNDX- $m$  are biased using a 3-dimensional 2-LVM. Let the parental population be  $G = \sum_{j=1}^2 e_j t_{i,j} + \epsilon$  as described in eq. 1. The variance  $\epsilon$  is sufficiently small, then PSDs are expected to be a linear combination of 2-dimensional latent variables  $e_i$ . When offspring is generated by UNDX-2( $m \geq k$ ), two PSCs are used to span 2-tablet space and the length  $D$  of SSC is correctly computed. Therefore, the generated offspring forms a 2-LVM similar to the parental population. To the contrary, when UNDX-1 is used( $m < k$ ), only one PSC is used and  $D$  is computed much larger than  $\epsilon$ . Then, the generated offspring



**Figure 4: Eigenvalues of the covariance matrix of parental population and its offspring generated by crossovers. Parental population is 20-dimensional 5-tablet(left) and 15-tablet(right) LVM. UNDX-1, 3, 5, 10, 19 were used for each LVMs.**

distributes isotropically not reflecting the 2-LVM parental population.

When the dimension is very high and  $m < k$ , offspring does not reflect the  $k$ -LVM of the parental population, that is, the sampling of UNDX- $m$  does not concentrate on the hopeful region. From above, it can be concluded that the number of parent  $m$  should be set at maximum,  $n - 1$ , from the viewpoint of the preservation of arbitrary  $k$ -LVMs.

### 3.4 SPX in $k$ -tablet Structures

Simplex Crossover (SPX)[8] generates offspring following the uniform distribution within an  $n$ -dimensional simplex defined by  $n + 1$  parents, where the simplex is larger  $\gamma$  times than the parental simplex. Fig. 2 shows the distribution of offspring generated by SPX. SPX fully preserves the covariance, hence, the preservation of arbitrary  $k$ -tablet structures is satisfied. However, mainly because of the following two reasons, RCGA using SPX converges considerably slow or fails to reach optimum, especially in high-dimension problems.

One is that the parental population easily shrinks to the dimension less than  $n$ . Once this happens, SPX cannot recover the lost dimension and this causes premature convergence. The other is that the parental population tends to converge in the region far from the optimum (premature convergence) because of the interpolative sampling bias of the crossover[6].

Although shrinking of population and the premature convergence can be avoided if the number of population is set at more than  $50n$  for SPX, the time to convergence becomes quite large in high dimensional functions (see section 5 in detail). UNDX- $m$  with  $n + 2$  parents which satisfies the preservation of statistics also faces with the same problem. Considering the preservation of statistics, UNDX- $n$  or SPX should be used, however, these crossovers are not preferable in terms of the convergence speed. The discussion about the premature convergence is shown in experimental section again. In next section, we propose a scalable RCGA to the dimensionality which can preserve  $k$ -LVM with small number of parents and population.

## 4. CROSSOVER ON LATENT VARIABLES

### 4.1 Our Proposal

To use UNDX- $m$  on  $k$ -tablet structures,  $m$  must be set at the value larger than  $k$  for satisfying the preservation of statistics, however, it is normally impossible to predict  $k$  of cost functions in advance. Although SPX or UNDX- $n$  fully preserves  $k$ -LVM, numerous number of population is required to avoid the premature convergence and the population shrinking. Therefore, following two conditions should be satisfied for efficient search in  $k$ -tablet structures : (1) Preservation of  $k$ -LVM at small  $m$ , (2) Avoiding premature convergence and population shrinking.

We separately use the PSDs from the SSDs and treat them as two crossovers. Although the distribution of offspring generated by PSDs fully preserves the statistics, the distribution shrinks in  $m$ -dimensional subspace, therefore, the parental distribution also tends to shrink and eventually the search of RCGA stagnates. To avoid this, we further use SSDs as another crossover to resist to the shrinkage. We call the former LUNDX- $m$  (UNDX using  $m$ -latent variables) and the latter EDX (Extrapolation-Directed Crossover). EDX has been proposed in [6] and performs as a mutation of RCGAs. The offspring of LUNDX- $m$   $\mathbf{x}^L$  and EDX  $\mathbf{x}^E$  is written in the form:

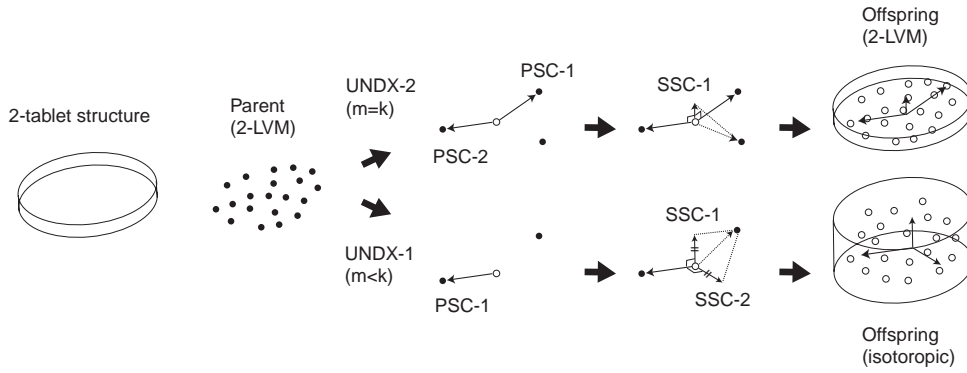
$$\mathbf{x}^L = \mathbf{p} + \sum_{i=1}^m w_i \mathbf{d}^i, \quad \mathbf{x}^E = \mathbf{x}^1 + \sum_{i=1}^{n-1} v_i D \mathbf{e}_i, \quad (3)$$

where the parents  $\mathbf{x}^1, \dots, \mathbf{x}^m$  are used in LUNDX- $m$  and 3 parents  $\mathbf{x}^1, \mathbf{x}^2, \mathbf{x}^3$  are used in EDX. In both crossovers, parents are chosen randomly from the population. By removing the SSDs from eq. 2, we obtain  $\mathbf{x}^L$ .  $\mathbf{x}^E$  corresponds to the SSDs of UNDX-1.

Next, we describe the procedure of RCGA using two crossovers. The procedure of RCGA is designed as a variants of MGG (Minimal Generation Gap) model [4]. MGG is superior in maintaining the diversity of the population and is applied widely to both combinatorial optimization and function optimization problems. In [1], it is reported that generation alternation model GGG, which adopts tournament selection, performs better in single peak functions in terms of convergence speed, however, we use MGG considering the global search performance.

#### [MGG using LUNDX- $m$ and EDX]

1. Create an initial population that is composed of  $p$  random vector drawn from a uniform distribution in a defined search region.
2. Choose  $m + 2$  vectors from the current population with probability  $p_L$  and jump to step 3. Otherwise, choose 3 vector and jump to step 5.
3. Generate a family by applying LUNDX  $n_{LUNDX}$  times and choose two offspring from the family that includes the parents. One has the lowest cost and the other is chosen by roulette selection.
4. Exchange the parents  $\mathbf{x}^1$  and  $\mathbf{x}^2$  for the selected two offspring and jump to step 8.
5. If  $f(\mathbf{x}^1) > f(\mathbf{x}^2)$ , then exchange  $\mathbf{x}^1$  for  $\mathbf{x}^2$ . Set  $t = 0$ .
6. Generate one offspring by EDX. If the fitness of the offspring is lower than that of  $\mathbf{x}^1$ , then exchange  $\mathbf{x}^1$  for the offspring and set  $t = t + 1$ .
7. If  $t > n_{EDX}$ , then jump to 8. Else jump to step 5.



**Figure 5: Generation of offspring where the parental population is 3-dimensional 2-LVM and the crossovers are UNDX-1 and UNDX-2. When  $m = 1$ ,  $k$ -LVM is not preserved because  $k > m$ .**

8. If the stop condition is satisfied, then stop. Else, jump to step 2.

EDX generates offspring in the region orthogonal to the latent variable space that the parents span. LUNDX- $m$  generates offspring almost in the latent variable space. On the contrary, the selection of EDX is carried out in a manner of deterministic local search to prevent the population from sprawling widely out of the latent variable space. Conversely, in LUNDX- $m$ , the parents are exchanged for the offspring having the best fitness and the offspring chosen by roulette selection for maintaining the diversity on the latent variable space.

**Preservation of  $k$ -LVM by LUNDX- $m$ :** Since PSDs used in LUNDX- $m$  are regarded as a linear combination of latent variables, offspring generated from LUNDX- $m$  shrinks in the  $m$ -dimensional latent variable space. However, it is theoretically proved that the entire offspring generated by the iteration of LUNDX- $m$  preserves  $k$ -LVM after the iteration of LUNDX- $m$  (proof is easily derived from the proof in [3]). The time complexity of SPX and UNDX- $n$  are  $O(n^2)$ , while the complexity of LUNDX- $m$  is  $O(mn)$ .

**Generation alternation model with two crossovers:** In LUNDX- $m$  which conducts its search in the latent variable space, the best and a roulette-selected vector are exchanged for the parents. This selection strategy aims at the maintenance of diversity in the latent variable space. To the contrary, EDX generates vectors orthogonal to the latent variable space. The improvement probability of EDX is low because the offspring is generated outside of the latent variable space, so if the roulette selection is used as well as LUNDX- $m$ , the convergence speed becomes very slow and waste computation time. Therefore, the parent is exchanged only when the cost is improved. By switching these two election strategies, the diversity on latent variable space is maintained while the shrinking of the population is avoided.

## 5. EXPERIMENTS

### 5.1 Setting

In this section, we perform three experiments to verify the performance of the proposed method. Conventionally, the dimensionality  $n$  of benchmark functions is set at around 20 to 30. In this paper, we use  $n = 20, 40, \dots, 320$  to study the performance on high dimensional functions. In experiment 1, the dimensionality is  $n = 20, 40, 60, 80, 100$ . We perform

a parameter survey of the population size and record the optimal convergence speed. In experiment 2, using  $k$ -tablet functions whose dimensionality  $n$  is more than 100, we compare the scalability to the dimensionality. In experiment 3, the global search performance is examined.

Definition of benchmark function is shown in table 1.  $f_1$  is a  $k$ -tablet function which is obtained by a linear transformation of sphere function.  $f_2$  is a 1-tablet (ridge) function whose ridge forms a parabola.  $f_3$  and  $f_4$  are multiple-peak functions which have numerous number of local optima. Initial population is generated randomly from uniform distribution in  $[-5.12, 5.12]^n$  ( $f_1, f_3$ ),  $[-2.048, 2.048]$  ( $f_2$  and  $[-512, 512]^n$  ( $f_4$ )).

Four crossovers, SPX, UNDX- $m$  with EDX(U+E), SPX with EDX(S+E) and LUNDX- $m$  with EDX(L+E) are compared. SPX fully satisfies the preservation of  $k$ -LVM but requires a large number of population. U+E are reported to reduce the sampling bias and perform well in 1-tablet functions but the performance in  $k$ -tablet functions is not known. S+E and L+E are expected to preserve  $k$ -LVM well with a small number of population. We do not compare UNDX- $m$  and U+E because it has been reported that U+E performs better than UNDX- $m$  in high dimensional ridge functions[6].

The minimum number of population  $p$  to avoid premature convergence is different by the tested functions and crossovers. For U+E,  $p$  is recommended to be set at  $n/4$  in [6]. By preliminary parameter survey, we use the minimum number of the population as  $p = n/4$  for L+E,  $p = 30n$  for SPX in  $n/4$ -tablet function,  $p = 20n$  for S+E in  $3n/4$ -tablet function and  $p = 2n$  for S+E. With the number less than these, we confirmed that premature convergence appears. The number of parents is set at  $m = \min\{25, 3p/4\}$  in U+E and L+E. For SPX,  $m$  is fixed as  $n + 1$ . The probability applying EDX is  $p_{EDX} = 0.5$  for U+E, S+E and L+E. The number of generated offspring is 100 for UNDX- $m$ , SPX and LUNDX- $m$  and 50 for EDX.

The procedure is terminated when one of the following conditions is satisfied: (1) the margin between the best cost of the population and the optimal cost reaches less than  $10^{-8}$  (optimal convergence); (2) the margin between the best cost of the population and the average cost over the whole population reaches less than  $10^{-10}$  (premature convergence); (3) the number of function evaluation reaches  $6.0 \times 10^8$  (time expired).



**Table 1: Definition and characters of benchmark functions. SP and MP mean single peak and multiple peaks respectively. Corner and Center means that the optimum locates at the corner and the center of the domain respectively.**

Function		Character	Domain	Optimum
1. $k$ -tablet	$f1(x) = \sum_{i=1}^k x_i^2 + \sum_{i=k+1}^n (100x_i)^2$	SP, $k$ -tablet	$(-\infty, \infty)$	$x_i = 0$
2. Rosenblock	$f2(x) = \sum_{i=1}^{n-1} (100(x_i^2 - x_{i+1})^2 + (x_i - 1)^2)$	SP, Corner bent 1-tablet	$(-\infty, \infty)$	$x_i = 1.0$
3. Rastrigin	$f3(x) = 10n + \sum_{i=1}^n [x_i^2 - 10 \cos(2\pi x_i)]$	MP, Center	$(-\infty, \infty)$	$x_i = 0.0$
4. Schwefel	$f4(x) = An + \sum_{i=1}^n x_i \sin \sqrt{ x_i }$ , A=418.9828873	MP, Corner	$[-512, 512]^n$	$x_i = -420.9688476$

**Table 2: Experiment 1: The number of function evaluation ( $\times 10^6$ ) until optimal convergence in  $k$ -tablet functions ( $k = n/4, n/2, 3n/4$ ) when the population size and the dimensionality is changed (the experiment is terminated by condition (1), the result is an averaged value over 20 iterations). “pc” means that the experiment is terminated by condition (2), (3) (premature convergence).**

f1, $k = n/4$												
crossover	LUNDX- $m$ +EDX			UNDX- $m$ +EDX			SPX			SPX+EDX		
dim.	p= $n/4$	p= $n/2$	p= $n$	p= $n/4$	p= $n/2$	p= $n$	p=30n	p=40n	p=50n	p=2n	p=3n	p=4n
$n = 40$	2.75	2.30	7.14	2.98	<b>2.04</b>	7.24	6.97	8.88	10.9	8.50	17.3	22.1
$n = 60$	6.28	<b>4.41</b>	15.2	6.08	4.53	16.5	13.8	16.7	20.7	15.7	37.7	45.8
$n = 80$	11.9	<b>7.36</b>	32.1	13.1	8.4	31.0	25.5	25.7	31.3	26.9	64.1	72.0
$n = 100$	23.8	<b>12.7</b>	46.9	27.0	17.7	46.7	pc	pc	43.9	39.9	92.6	99.4
f1, $k = 3n/4$												
crossover	LUNDX- $m$ +EDX			UNDX- $m$ +EDX			SPX			SPX+EDX		
dim.	p= $n/4$	p= $n/2$	p= $n$	p= $n/4$	p= $n/2$	p= $n$	p=20n	p=25n	p=30n	p=2n	p=3n	p=4n
$n = 40$	<b>1.68</b>	3.80	7.70	1.83	3.91	7.60	3.82	4.72	5.76	14.4	22.1	30.5
$n = 60$	3.49	7.57	17.2	<b>3.46</b>	7.61	16.6	7.19	8.73	10.5	31.8	50.4	68.8
$n = 80$	<b>5.91</b>	11.4	27.3	6.09	11.7	26.9	11.7	13.5	16.1	52.7	85.5	118
$n = 100$	<b>8.85</b>	16.6	56.1	9.21	16.5	36.6	pc	19.3	22.5	78.1	125	178

## 5.2 Experiment 1: Low Dimensionality

In experiment 1, the optimal convergence speed of U+E, SPX, S+E and L+E are compared in  $k$ -tablet function. The population size of each setting is determined considering the minimum number of population in previous section. In table 2, the number of function evaluation ( $\times 10^6$ ) until optimal convergence is shown. In this experiment, all trials are terminated by condition 1(optimal convergence) or no trial reaches to the optimum (premature convergence), therefore, the number of optimal convergence is not written.

The table shows that U+E and L+E converges 2 to 10 times faster than SPX and S+E. L+E is slightly faster than U+E in most of the setting but the difference is small. In L+E and U+E, the minimum population size  $p = n/4$  shows the best performance in  $3n/4$ -tablet functions, however,  $p = n/2$  is better in  $n/4$ -tablet functions. Considering the optimization of arbitrary  $k$ -tablet functions, we adopt  $p = n/2$  in the latter experiments because the smaller  $k$  is, the slower the convergence speed generally is.

## 5.3 Experiment 2: High Dimensionality

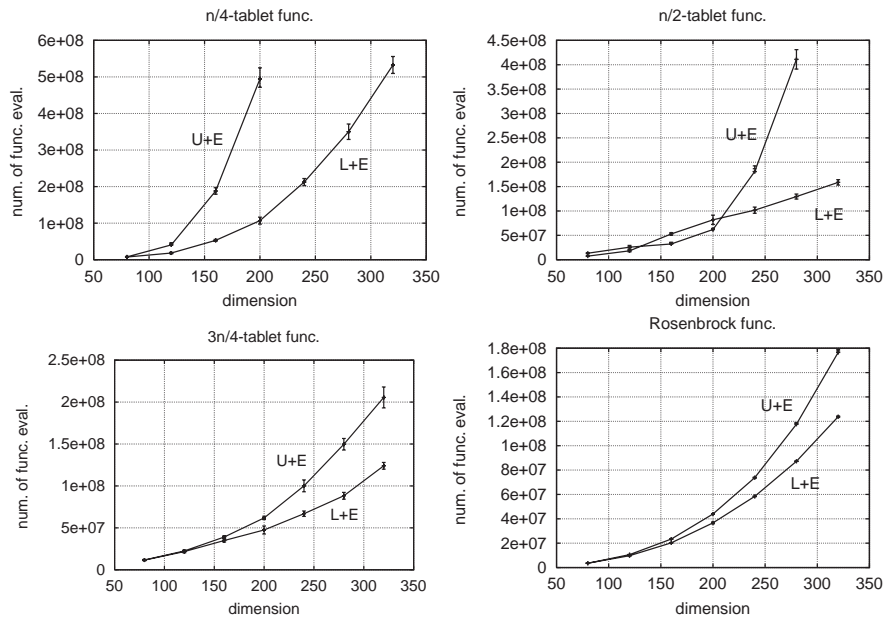
In this experiment, optimal convergence speed when  $n$  is varied from 80 to 320 in function  $f1$  and  $f2$ . From experiment 1, it is clarified that L+E and U+E performs better than SPX and S+E in  $k$ -tablet functions, so only L+E and U+E is evaluated here. The population size is  $p = n/2$  and  $k = n/4, n/2, 3n/4$ . The other conditions are the same in experiment 1. Fig 6 shows the average of required number of function evaluation to the optimal convergence. The result is an average of 10 trials in  $n < 200$  and 5 trials in  $n \geq 200$ .

In  $f1$  of  $k = 3n/4$  and  $f2$ , L+E performs better than U+E but the difference is not large. On the other hand, in  $f1$  of small  $k$ , L+E is 2 to 5 times faster than U+E at  $n \leq 200$ . Although experiments of U+E are terminated by the condition (3) in  $n \geq 200$ , it is expected that L+E is faster than U+E 10 to 100 times. We compare L+E and U+E in ellipsoid function and different power function[2], which have  $k$ -tablet like landscape and confirmed that L+E converges 15% to 20 % faster than U+E.

In this experiment, the number of parent  $m = \{25, 3p/4\}$  is used. This means that  $m$  is limited even if  $k$  is larger than 25. In this setting, when  $n$  is larger than 100, the preservation of  $k$ -LVM is not satisfied in U+E regardless of  $k$  and  $n$  because  $m < k$ . The volume of  $k$ -tablet space with small  $k$  relatively decreases exponentially to  $n$ , therefore, the convergence speed of crossovers which do not satisfy the preservation of  $k$ -LVM is considerably slow. Repeatedly, when large  $m$  is used,  $k$ -LVM is preserved but U+E with large  $m$  also requires large population and the convergence speed is slow same as shown in the result of SPX in experiment 1. Therefore, it is clear that L+E, which preserves arbitrary  $k$ -LVM with small  $m$  works effectively. From these results, we can conclude that that the proposed method performs better than conventional methods in  $k$ -tablet functions.

## 5.4 Experiment 3: Global Search Performance on Multiple-peak Function

In this experiment, we compare the global search performance by using multiple-peak functions  $f3$  and  $f4$ . The dimensionality is  $n = 20$  in  $f3$  and  $n = 10$  in  $f4$ . The popu-



**Figure 6: Experiments 2: The number of function evaluation until optimal convergence in  $f_1(k$ -tablet function,  $k = n/4, n/2, 3n/4$ ) and  $f_2$ (Rosenbrock function) for different dimensionality. The result is an averaged value over 10 iterations when  $n < 200$  and 5 iteration when  $n > 200$ . The error bar shows the standard deviation.**

lation size  $p = 300$  and the number of the parent is  $m = 10$  for UNDX- $m$  and LUNDX- $m$ . In  $f_3$ , U+E and L+E found the global optimum in 30 and 29 times over 30 trials respectively. In  $f_4$ , U+E and L+E found the global optimum in 25 and 23 times over 30 trials respectively. The convergence speed of L+E is slightly faster than U+E. The global search performance of U+E is slightly better than L+E but the difference is small. L+E tends to lose the diversity because the sampling of LUNDX- $m$  is limited on  $k$ -tablet space, however, EDX contributes to the maintenance of diversity and the global search performance does not degrade largely.

## 6. $K$ -TABLET STRUCTURE ON LENS DESIGN PROBLEMS

### 6.1 Landscape Analysis Using PCA

The lens design problem is known as a difficult problem because of high dimensionality, non-separability and numerous number of local optima. We focus on  $k$ -tablet structures in this problem and show that proposed GA also performs well in this problem. The lens design problem is described as follows. We consider to search the curvature of lens surfaces  $c_i$  and thickness between two surfaces  $d_i$  to obtain a lens that has good image forming capabilities (see Figure 7). A focal length  $f$ , an F/number  $F$  and a field size  $2w$  is given as a specification. Because the curvature of the last surface and the distance of it to the image surface are modified to meet a required focal length  $f$ , an  $N$ -element lens design problem can be considered as a function optimization problem with  $4N - 2$  parameters [4].

We use two criteria, the distortion  $D$  and the resolution  $R$ . Both  $R$  and  $D$  are evaluated using spot diagram by performing ray tracing. The distortion  $D$  is defined as the distance between the image point of a principal ray and the

ideal image point. The resolution  $R$  is the size of the blur of rays surrounding a principal ray. See [4] in detail.

Most of individuals in a population in the middle of the search are considered to have similar cost value. Therefore, the actual shape of the contour around this cost is expected to be estimated by investigating the form of the population. Using this property, we can know whether the contour surface forms  $k$ -tablet structure or not by applying PCA to the population and comparing the eigenvalues of each component.

We analyze the landscape of the 4-element lens design problem. The specifications are  $f = 50\text{mm}$ ,  $F/2.0$ ,  $2w = 46.0$ . The population is collected by the iteration of RCGA (UNDX-5+EDX, MGG,  $p = 50, c = 100$ ) 130 times and save all individuals. Then, we classify them into 3 clusters  $S_1, S_2, S_3$  such that the cost of individual belonging to  $S_1, S_2, S_3$  is in  $[1.00, 2.00)$ ,  $[0.50, 1.00)$ ,  $[0.25, 0.50)$ , respectively.

Figure 8 (above right) shows the projection of  $S_2$  onto  $c_3 - c_5$  plane. As shown in this figure, the population is composed of several clusters. Each cluster corresponds to a specific type of a lens. Two typical lenses and clusters corresponding to these lenses are shown. This suggests that there exist several global peaks on the landscape and each peak corresponds to a specific type of lens. PCA is valid only when the data distribution is isolated as a cluster. Therefore, we apply  $k$ -means algorithms ( $k=6$ ) and divide the set into 6 clusters. Figure 8 (bottom, left and right) show the distribution of the eigenvalues of 2 clusters of 6. The results indicate that the largest eigenvalue is almost  $10^4 \sim 10^5$  times as large as the smallest one. We confirm that each cluster forms a kind of  $k$ -tablet distribution that is, each peak forms a  $k$ -tablet like structure.

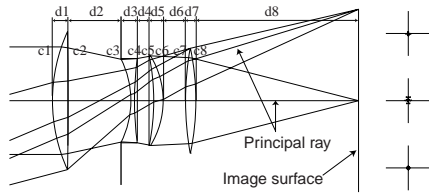


Figure 7: An example of lens system

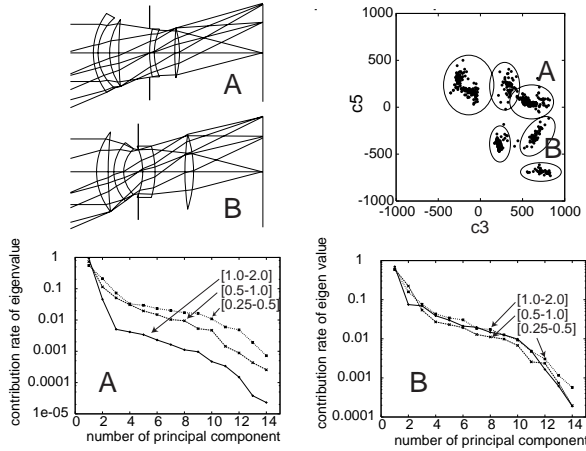


Figure 8: Typical two examples of lens belonging to cluster A and B(above left); Projection of  $S_2$  onto  $c_3 - c_5$  plane(above right); Distribution of eigenvalues of cluster A and B obtained by PCA(below)

## 6.2 Application to Lens Design Problems

It is expected that LUNDX- $m$  performs better than other crossovers because of  $k$ -tablet like structures. We compare the performance of RCGA using U+E and L+E in 4-element problems. For both RCGAs, the population size  $p = 200$  and the number of the parent  $m = 5$ . To see the performance in different lens type, we prepare two regions to initialize the population. The population is generated as follows: (1)  $\mathbf{x}_A, \mathbf{x}_B \in [-1000, 1000]^n$  is generated randomly; (2) the initial population is generated uniform randomly in  $S_A = [x_A^i - 500, x_A^i + 500]^n$  and  $S_B = [x_B^i - 500, x_B^i + 500]^n$  (initializing region). 5 trials are conducted in each initializing region. The convergence curve of L+E and U+E is shown in fig. 9. The RCGA was terminated after  $3.0 \times 10^6$  evaluations.

The figure shows L+E performs better in terms of both search speed and the quality of the obtained lens in comparison with U+E. These results fit well with results in section 5.

## 7. CONCLUSION

In this paper, we paid particular attention to the fact that the traditional RCGA degrades in  $k$ -tablet structures. Thus, we proposed a new RCGA using two crossovers LUNDX- $m$  and EDX, where PSDs and SSDs are separately used. Experimental results show that our proposal performs robustly with a small population size and converges to optimum quickly on  $k$ -tablet functions. Moreover, we apply our method to lens design problems and achieve good per-

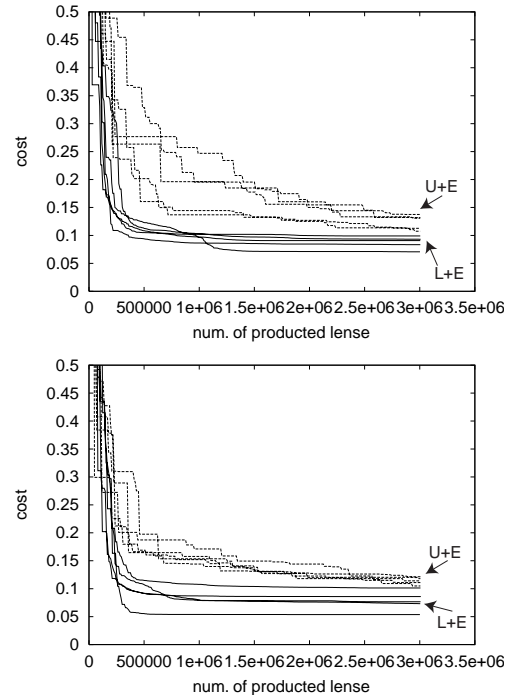


Figure 9: Convergence curve on lens design problems on two different type of lenses(results of 5 trials are shown respectively)

formance in terms of the search speed and the quality of obtained lens. As dimensionality  $n$  grows, it is expected that this tendency will become remarkable. Applying our method to larger lens systems is our future work.

## 8. REFERENCES

- [1] Deb., K. and Anand A., and Jochi, D., A Computationally Efficient Evolutionary Algorithm for Real-Parameter Optimization, *Evolutionary Computation*, Vol, 10, Issue 4, pp. 371-395 (2002).
- [2] Hansen, N. and Ostermeier, A. Completely Derandomized Self-Adaptation in Evolution Strategies, *Evolutionary Computation*, 9(2) : pp. 159-195 (2001).
- [3] Kita, H., Ono, I. and Kobayashi, S. (1999), Multi-parental Extension of the Unimodal Normal Distribution Crossover for Real-coded Genetic Algorithms, *Proc. of Conference on Evolutionary Computation 1999*, pp. 1581-1587(1999).
- [4] Ono, I. Kobayashi, S. and Yoshida, K., Optimal lens design by real-coded genetic algorithms using UNDX, *Comput. Methods Appl. Mech. Engrg.* 186: pp.483-497 (2000).
- [5] Oyman, A. I., Bayer, H. G. and Scherfel, H. P., Analysis of the  $(1, \lambda)$ -ES on the Parabolic Ridge, *Evolutionary Computation*, 8(3) : pp.249-265 (2000).
- [6] Sakuma, J. and Kobayashi, S., Extrapolation-Directed Crossover for Real-coded GA: Overcoming Deceptive Phenomena by Extrapolative Search, *Proc. of Congress on Evolutionary Computation*, pp. 655-662 (2001).
- [7] Tipping, M. E. and Bishop, C. M. , Mixtures of Probabilistic Principal Component Analyzers, *Neural Computation*, vol. 11, pp. 435-474 (1998).
- [8] S. Tsutsui., Yamamura, M. and Higuchi, T., Multi-parent Recombination with Simplex Crossover in Real Coded Genetic Algorithms, *Proc. of the Genetic and Evolutionary Computation Conference*, Vol. 1, pp. 657-664 (1999).