# Increasing Robustness of Genetic Algorithm

**Jiangming Mao, Kotaro Hirasawa, Jinglu Hu and Junichi Murata**
Graduate School of Information Science and Electrical Engineering, Kyushu University,
6-10-1, Hakozaki, Higashi-ku, Fukuoka 812-8581, Japan
mao@cig.ees.kyushu-u.ac.jp

## Abstract

Genetic algorithms are often well suited for optimization problems because of their parallel searching and evolutionary ability. Crossover and mutation are believed to be the main exploration operators in GA. In this paper, we focus on how crossover and mutation work in GA and investigate their effect on bit's frequency of the population. To increase robustness against uncertainty of GA, a new recombination method based on bit's frequency of the population and a new robust generation strategy were proposed. The proposed methods were tested on the problem with many local minima. Simulation results demonstrate the effectiveness of the proposed methods.

## 1 Introduction

Genetic algorithm (GA) is a random searching method with some special features. One feature is that GAs are versatile evolutionary computation techniques largely based on the principle of survival of the fittest [1]. Another is the genetic operators such as crossover and mutation. When using GA for solving a given problem, the user has to design so many parts to make GA effective, such as the number of population, population size, mutation rate, crossover rate, selection pressure and selection scheme. However, GA can not always get good solutions we want, because it is difficult for users to design an effective GA which is a random searching method.

A prevalent method in GA is to assign survival probabilities to corresponding individuals and tune the probabilities to obtain the balance between exploration and exploitation[2]. In GA, the crossover and mutation are believed to be the main exploration operators in the working of GA as an optimization tool. In this paper, we focus on the variance of the distribution of the individuals on a hyper-plane, through a new way to investigate how the mutation and crossover work in GA. Furthermore, a new **rec**ombination method based on **b**it's **f**requency (RCBF) was proposed, which can make the population distribute more uniformly than the "conventional" crossover such as one point crossover, two point crossover and uniform crossover. In addition, because all messages from the population are stored in bit's frequency, a new **r**obust **g**eneration **s**trategy (RGS) is proposed where the $t+m$ th generation is determined not only by the $t+m-1$ th generation but also by generations from the $t$ th to the $t+m-2$ th. According to the simulation results, we can find GA by using RCBF and RGS can search for the solutions more robustly than "conventional" GA, especially when the feasible solution space is very large.

This paper is organized as follows. Next section is about a new way of analyzing crossover and mutation of GA. Section 3 introduces RCBF and give some simulation results. Section 4 introduces RGS and give some simulation results. The last section offers concluding remarks and future perspectives.

## 2 A New Way of Analyzing GA

The GA studied in this paper is the one similar to Simple Genetic Algorithm defined in [2].

### 2.1 Mathematical description

A $k$ th binary individual $X_k$ in a population can be given by

$$X_k = (x_k^1, \ldots, x_k^j, \ldots, x_k^L), \tag{1}$$

where $L$ is the length of the binary individual, $x_k^j$ stands for the $j$ th bit of the $k$ th individual. A population $\vec{X}$ can be defined as

$$\vec{X} = (X_1, \ldots, X_k, \ldots, X_N), \tag{2}$$

where $N$ is the population size. The feasible space of bit $x_k^j$ is $\{0,1\}$. The feasible space of the individual $X_k$ is $\{0,1\}^L$.

**Definition** 2.1 (Bit's frequency) Let $f_{\vec{X}}^j$ be the $j$ th bit's frequency of the population $\vec{X}$, where

$$f_{\vec{X}}^j = \frac{1}{N} \sum_{k=0}^{N} x_k^j. \tag{3}$$

The feasible space $S_f$ of $f_{\vec{X}}^j$ is $[0,1]$. The bit's frequency string $F_{\vec{X}}$ can be given by

$$F_{\vec{X}} = (f_{\vec{X}}^1, \ldots, f_{\vec{X}}^j, \ldots, f_{\vec{X}}^L), \tag{4}$$

where the feasible space $S_f^L$ of the bit's frequency string is $[0,1]^L$.

If the population is distributed in $Z^L$, the population $\vec{X}$ is a set of the vertex of the unit-box with $L$ dimensions. The bit's frequency string $F_{\vec{X}}$ can be represented in $R^L$. We can see a population is a dynamical structure with a centre of gravity $F_{\vec{X}}$ in $R^L$. To investigate the effect of mutation and crossover, we will do some research on the variance of the centre of gravity of the population.

### 2.2 Crossover Operator

The crossover operator $T_c$ is a very complex operator to recombine the gene of each individual in the population. There exist a number of crossover operators in the GA literature, such as one point crossover, two point crossover and uniform crossover. According to the quality of crossover, we know the crossover operators don't change the bit's frequency string. To investigate the effect of the crossover operator, let us see the next definition.

**Definition** 2.2 If $F_{\vec{X}}{=}F_{\vec{Y}}$, we can say the population $\vec{X}$ is similar to the population $\vec{Y}$, denoted by $\vec{X} \sim \vec{Y}$.

Because we can not derive $\vec{X} = \vec{Y}$ ($\vec{X}$ and $\vec{Y}$ are the same) from $\vec{X} \sim \vec{Y}$, the crossover operators can change the population from one case to another with the same bit's frequency string.

**Definition** 2.3 (Distribution state function) we use a two-order function to show the distribution state $E_{\vec{Y}}$ of the population $\vec{Y}$ as follows,

$$E_{\vec{Y}} = \frac{N^2_{(0,\ldots,0)} + N^2_{(0,\ldots,1)} + \ldots + N^2_{(1,\ldots,1)}}{N^2}, \tag{5}$$

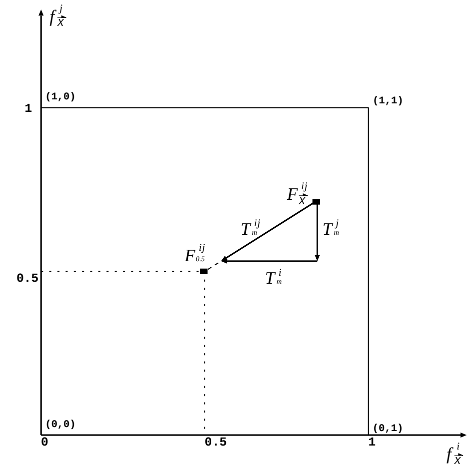


Fig. 1. Mechanism of mutation in $F_{\vec{X}}$.

where $N_{(\ldots)}$ is the overlapping number of the individual $(\ldots)$, and $N_{(0,\ldots,0)} + N_{(0,\ldots,1)} + \ldots + N_{(1,\ldots,1)} = N$.

If the bit's frequency string $F_{\vec{Y}}$ of the population $\vec{Y}$ is determined, using the following $N_{(y^1,\ldots,y^j,\ldots,y^L)}$

$$N_{(y^1,\ldots,y^j,\ldots,y^L)} = \lfloor (\prod_{j=1}^{L} \hat{f}_{\vec{Y}}^j)\ N + 0.5 \rfloor, \tag{6}$$

where

$$\hat{f}_{\vec{Y}}^j = \begin{cases} f_{\vec{Y}}^j & \text{if } y^j = 1 \\ 1 - f_{\vec{Y}}^j & \text{if } y^j = 0 \end{cases}$$

then $E_{\vec{Y}}$ has the minimum value, denoted by $E_{\min}$. Furthermore, as $E_{\min}$ is determined by the bit's frequency string $F_{\vec{Y}}$, when $F_{\vec{Y}} = F_{0.5}$ (where $F_{0.5} = (0.5, \ldots, 0.5)$), $E_{\min}$ will be the most minimum.

Supposing we use $T_c$ to make the population $\vec{X}$ crossover $d$ times where $d = 1, 2, \ldots$, then we can give $E_{\vec{X}} \to E_{\min}$ when $d \to \infty$. In other words, the crossover operator has an ability to make the population distribute uniformly without changing the bit's frequency string.

### 2.3 Mutation Operator

The mutation operator is a force $T_m$ (a vector quantity) to maintain the diversity in the population and is used with a small probability, $p_m$. To give the direction and strength of the mutation force, we describe the population $\vec{X}$ onto the plane, for example, $(f_{\vec{X}}^i, f_{\vec{X}}^j)$ plane shown in Fig.1, where $f_{\vec{X}}^i$ and $f_{\vec{X}}^j$ are the lateral and vertical coordinates. In Fig.1, $F_{0.5}^{ij}$ is the two dimensional point of $F_{0.5}$ , $F_{\vec{X}}^{ij}$ is the two dimensional point of $F_{\vec{X}}$, $T_m^{ij}$ is the two dimention vector quantity of $T_m$ on the plane $(f_{\vec{X}}^i, f_{\vec{X}}^j)$, $T_m^i$ and $T_m^j$ are the component quantities of $T_m$ on the $f_{\vec{X}}^i$ and $f_{\vec{X}}^j$ coordinates respectively. We can easily give $T_m^i$ and $T_m^j$ as follows,

$$T_m^i = \frac{N_{(x^i=1)} - N_{(x^i=0)}}{N}\ p_m,$$
$$T_m^j = \frac{N_{(x^j=1)} - N_{(x^j=0)}}{N}\ p_m,$$

where $N_{(*)}$ is the number of individuals of the population $\vec{X}$ where $x^i$ or $x^j$ is equal to 0 or 1, therefore $N_{(x^i=1)} + N_{(x^i=0)} = N_{(x^j=1)} + N_{(x^j=0)} = N$. So we can easily give the strength of $T_m^{ij}$ as follows,

$$\begin{aligned} |T_m^{ij}| &= \sqrt{\left(\tfrac{N_{(x^i=1)}-N_{(x^i=0)}}{N}\right)^2+\left(\tfrac{N_{(x^j=1)}-N_{(x^j=0)}}{N}\right)^2}\, p_m \\ &= 2\ |\overrightarrow{F_{\vec{X}}^{ij}F_{0.5}^{ij}}|\ p_m, \end{aligned}$$

where $\overrightarrow{F_{\vec{X}}^{ij}F_{0.5}^{ij}}$ is a vector from the point $F_{\vec{X}}^{ij}$ to $F_{0.5}^{ij}$, $|\overrightarrow{F_{\vec{X}}^{ij}F_{0.5}^{ij}}|$ is the length of the vector $\overrightarrow{F_{\vec{X}}^{ij}F_{0.5}^{ij}}$. The direction of $T_m^{ij}$ can be easily demonstrated to be the same as the direction of the vector $\overrightarrow{F_{\vec{X}}^{ij}F_{0.5}^{ij}}$. Generally, we can easily give the strength of $T_m$ as follows,

$$|T_m| = 2\ |\overrightarrow{F_{\vec{X}}F_{0.5}}|\ p_m, \tag{7}$$

where $|\overrightarrow{F_{\vec{X}}F_{0.5}}|$ is the distance between point $F_{\vec{X}}$ and $F_{0.5}$. The direction of $T_m$ is from point $F_{\vec{X}}$ to $F_{0.5}$.

According to Eq.(7), we can see mutation operator can change the bit's frequency string, where the strength of the mutation force is changed proportionally along with the convergence status (represented by $|\overrightarrow{F_{\vec{X}}F_{0.5}}|$) of the population and the direction of the mutation force is always from the point $F_{\vec{X}}$ to $F_{0.5}$. In other words, the mutation operator can change $E_{\min}$ which is determined by the bit's frequency string.

Furthermore, the mutation operator has another ability which is the same as the crossover operator. For example, if the bit's frequency string $F_{\vec{X}}=F_{0.5}$ and $E_{\vec{X}} > E_{\min}$, mutating the population $\vec{X}$ infinite times, $E_{\vec{X}}$ should be $E_{\min}$ without changing the bit's frequency string. Generally, this ability of the mutation operator exists in the case even when $F_{\vec{X}} \neq F_{0.5}$ and is smaller and smaller along with the concentration of the population.

There exist two kinds of abilities of the mutation operator, so we can separate the mutation operator into two parts: the first part which is determined by $|N_{(x^*=1)} - N_{(x^*=0)}|$ can change the the bit's frequency string while the second part which is determined by $\min\{N_{(x^*=1)}, N_{(x^*=0)}\}$ can make the population distribute uniformly without changing the bit's frequency string.

### 2.4 Concentration of the population

In a searching process by using GAs, the variance of the individuals' fitness is reduced due to two factors. One factor is selection pressure producing multiple copies of fitter population members while the other factor is independent of population member's fitness and is due to the stochastic nature of the selection operator, –genetic drift.[3]

Under the operation of selection, the fitter member of the population have higher chance of producing more offspring than the less member. If the selection pressure is greater than the mutation and crossover force, selection pressure makes all individuals of the population concentrate to the optimal points. We can separate selection methods into two main categories: using ranking methods [4][5] and not using ranking methods. The selection pressure without ranking methods is determined by the difference of the individuals' fitness, so it changes along with the evolutionary process. The selection pressure with ranking methods doesn't change along with the difference of the individuals' fitness. So the selection pressure with ranking methods can be more easily controlled than without ranking methods. But it takes much time to calculate the rank of each individual. Genetic drift makes the population concentrate randomly. The effect of genetic drift is not shown very clearly when the objective function is a unimodal function. But for multimodal functions, genetic drift should make the population concentrate to one of the optimal solutions randomly.

Anyway, selection pressure and genetic drift make the population concentrate. In other words, they make the $|\overrightarrow{F_{\vec{X}}F_{0.5}}|$ and $E_{\vec{X}}$ large.

## 3 Recombination Method based on Bit's Frequency

### 3.1 Species and Sampling

From the previous section, when $\vec{X} \sim \vec{Y}$ and $E_{\vec{Y}} = E_{\min}$, the mutation (the second part) and crossover operators make the population $\vec{X}$ approach $\vec{Y}$. In other words, the population $\vec{Y}$ is stabler than the population $\vec{X}$. Using this, a new recombination method stated in 3.2 is proposed.

**Definition** 3.1 (Species) A species can be defined as: a group of individuals that 1)actually or potentially

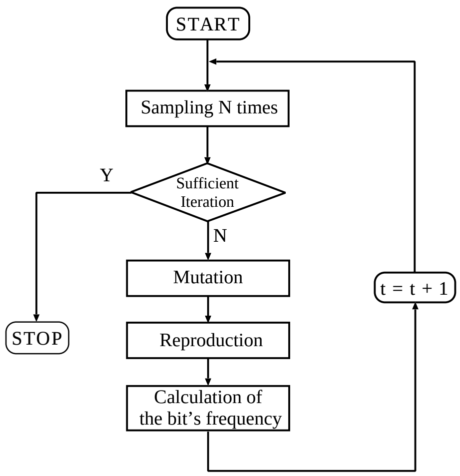


Fig. 2. Structure of the proposed algorithm based on bit's frequency.

interbreed with each other but not with other groups, 2)the distinction from other groups is the bit's frequency string $F$, where $F = (f^1, \ldots, f^j, \ldots, f^L)$, $f^j$ is the frequency when '1' appears in the $j$ th bit of the species

**Definition** 3.2 (Sampling) Sampling is an operator such as getting sample individuals from a species. This operator is given as follows: each bit of sample individuals is determined randomly according to the bit's frequency string $F$ of the species.

According to the definitions, a species is a population with a certain bit's frequency string. The individuals of the species can crossover with each other but can not do with other species. Each bit's of the individual $X$ of the species can be determined by this bit's frequency. It means the distribution of the individuals can satisfy Eq.6 and is not changed by crossover and the second part of mutation because $E_{\vec{X}}$ is minimum.

### 3.2 Flow of the proposed method

An simple genetic algorithm by using RCBF is shown in this subsection. The basic structure is shown in Fig.2, where the initial value of the bit's frequency string $F$ is $F_{0.5}$. One iteration at the $t$ th generation can be described as follows: 1)after sampling $N$ times according to the bit's frequency string $F_{(t)}$ we can get a population $\vec{X}_{(t)}$ with $N$ members; 2)after mutation and reproduction we can get a population $\vec{X'}_{(t)}$; 3)we can calculate the bit's frequency string $F_{\vec{X'}_{(t)}}$ of the population $\vec{X'}_{(t)}$ and set the bit's frequency string for the next generation.

In fact, we can consider RCBF as the strongest crossover, because it can make the population $\vec{X'}_{(t)}$ to be the population $\vec{X}_{(t+1)}$ which is distributed the most uniformly. It means RCBF can search more points than "conventional" crossover method such as the uniform crossover method. In other words, although the feasible search points of RCBF and uniform crossover are the same, RCBF can search for more of them than the uniform crossover. Especially when the feasible solution space is very large, doing more searching is very useful to increase the searching ability and the robustness of GA. Furthermore, because all messages from the environment are stored in the bit's frequency, sometimes, it is very useful to use RCBF in order to decrease the memory and time required for calculation.

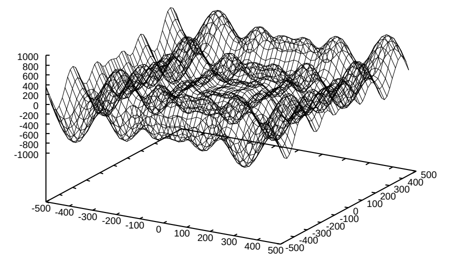

Fig. 3. The two-dimensional version of $f_8$.

### 3.3 Experiments

Generalized Schwefel's Problem which was examined in [6]-[7] is used in our experimental studies.

$$
\begin{aligned}
\min f_8(x) &= -\sum_{i=1}^{K}(x_i \sin(\sqrt{|x_i|})), \\
\text{where} \quad & K = 1, 2, \ldots, 30 \\
& -500 < x_i < 500
\end{aligned}
$$

This function is a multimodal function with many local minima, where the number of local minima increases exponentially as the dimension of the function increases like $7^K$. The global minimal function's value is $K \times 418.98289$. Fig.3 shows the two-dimensional version of $f_8$. To analyze the genetic algorithm by using the species concept, we can do some comparisons of the proposed method with the uniform crossover method.

#### 3.3.1 Parameter Values

- Population size: Since the problem dimensions are high, we choose a moderate population size $N$=200;
- Representation: Each variable has 30 bits, so the length of the individuals is $30 \times K$.

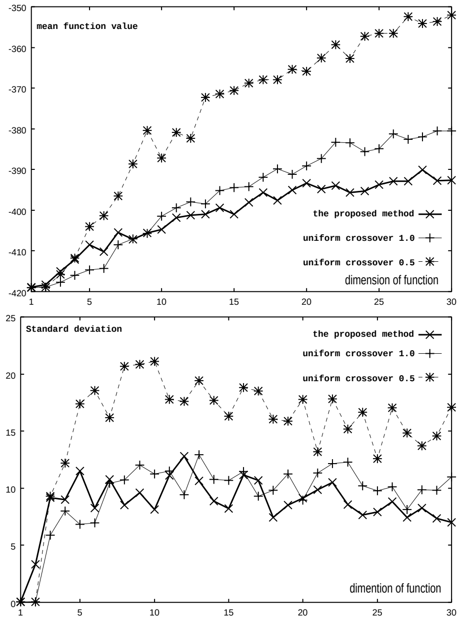


Fig. 4. Comparison between the proposed recombination method and the uniform crossover method in terms of the mean and the standard deviation of function value.

- Crossover rate: We set crossover rate 1.0 and 0.5 for the uniform crossover method respectively.
- Mutation probability: We choose $p_m = \frac{1}{L}$.
- Selection pressure: We use the nonlinear ranking method[5] where the selection probability of the $k$ th individual can be calculated as $p_k = c \times (1-c)^{i-1}$, $i$ is the rank of the $k$ th individual. We set the parameter $c = 0.05$.
- Iteration: The stopping generation is $\lfloor 50 \times \sqrt{K} + 0.5 \rfloor$.

### 3.3.2 Discussions

We performed 50 independent runs for the proposed method and uniform crossover method from $K = 1$ to $K = 30$ and recorded 1)mean function value (the mean value of the best individual of the last generation over 50 runs) and 2)the standard deviation of function value (the standard deviation of the best individual of the last generation over 50 runs). Fig.4 shows the simulation results. The upper part shows the mean function value $\bar{f}$ where the lateral coordinate is the dimension of the test function, while the lower part shows the standard deviation of the function value $\sigma_f$.

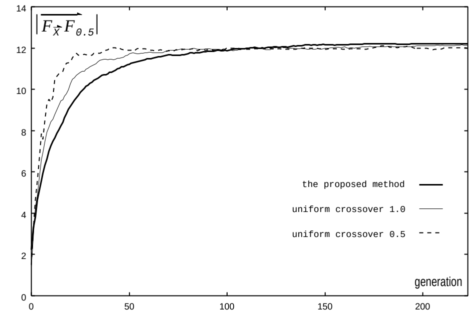


Fig. 5. Comparison of the convergence speed between the proposed recombination method and the uniform crossover method.

They can be calculated as follows:

$$\bar{f} = \frac{1}{50}\sum_{i=1}^{50}\frac{f_i}{K},$$

$$\sigma_f = \sqrt{\frac{1}{50}\sum_{i=1}^{50}(\frac{f_i}{K} - \bar{f})^2}.$$

According to these results, we can see

- when the dimension of the function is large, the mean function value of the proposed method is smaller than that of the uniform method with crossover rate 1.0, followed by that of the uniform crossover with crossover rate 0.5. It means that the search ability of RCBF is strongest compared with the uniform crossover. The standard deviation of RCBF is smaller than that of the uniform method with 1.0 crossover rate, followed by that of the uniform method with 0.5 crossover rate. It means that RCBF can increase the robustness against uncertainty of GA.
- when the dimension of function is small, the mean function value and the standard deviation of the proposed method is larger than those of the uniform crossover method.

Fig.5 shows the simulation results of the convergence speed which was randomly selected when $K = 20$, where the lateral coordinate is generation and the vertical coordinate is $|\overrightarrow{F_{\vec{X}}F_{0.5}}|$. According to these results, we can see that the population concentrating rates of the proposed method is slower than that of the uniform crossover with 1.0 crossover rate, followed by that of the uniform crossover with 0.5 crossover rate.

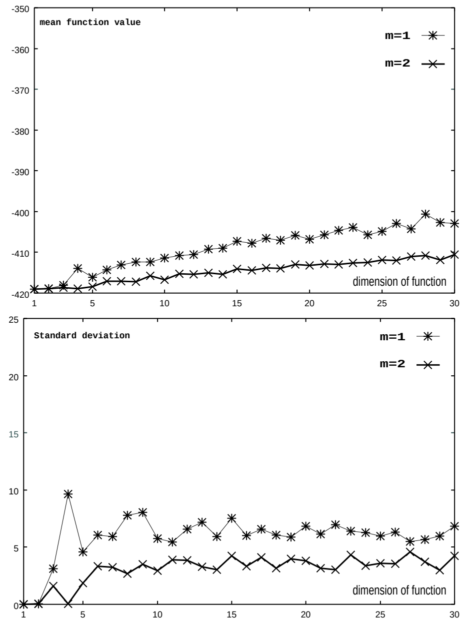


Fig. 6. Mean and standard deviation by using RGS under $m = 1, 2$ respectively.

# 4 Robust Generation Strategy (RGS)

## 4.1 Description of RGS

Because all messages from the population are stored in the bit's frequency, in order to increase robustness against uncertainty of GA, we calculate the bit's frequency string $F_{(t+m)}$ of the $t + m$ th generation as follows,

$$F_{t+m} = (f^1_{t+m}, \ldots, f^j_{t+m}, \ldots, f^L_{t+m}), \quad (m = 1, 2, 3, \ldots)$$

where

$$f^j_{t+m} = \frac{1}{m+1}\,(f^j_{\vec{X}'(t+m-1)} + \sum_{i=0}^{m-1} f^j_{t+i}) \qquad (8)$$

$f^j_{\vec{X}'(t+m-1)}$ means the $j$ th bit's frequency of the population $\vec{X}'$ at the $t+m-1$ th generation. Eq.8 means the bit's frequency string at the $t+m$ th generation is determined not only by the bit's frequency string $F_{\vec{X}'(t+m-1)}$ but also by the bit's frequency string from the $t+m-1$ th to the $t$ th generation. This method is named robust generation strategy(RGS).

## 4.2 Reason of Robustness

To investigate the effect of RGS, let us see a special case where $m = 1$. If $m = 1$, Eq.8 can be described as follows,

$$f^j_{t+1} = \frac{1}{2}\,(f^j_{\vec{X}'(t)} + f^j_t). \qquad (9)$$

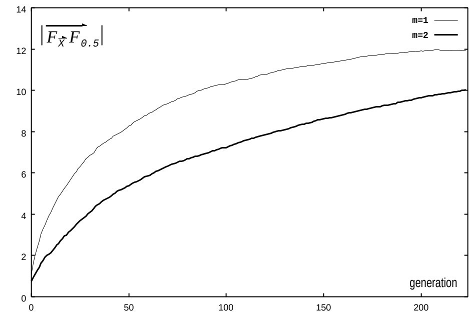


Fig. 7. Population concertrating rates by using RGS under $m = 1, 2$ respectively.

Eq.9 is a recursion formula and can be easily converted into as follows,

$$\begin{aligned} f^j_{t+1} &= \frac{1}{2} f^j_{\vec{X}'(t)} + \frac{1}{2^2} f^j_{\vec{X}'(t-1)} + \cdots + \frac{1}{2^t} f^j_{\vec{X}'(1)} \\ &+ \frac{1}{2^t} f^j_1 \end{aligned} \qquad (10)$$

According to Eq.10, we can see the effect of generations from the first to the $t$ th on the $t+1$ th generation. This is reason why RSG can make GA search for solutions robustly against uncertainty than "conventional" GA. When $m \geq 2$, the relationship between $f^j_{t+m}$ and $f^j_{\vec{X}'(t+m-1)}, \cdots, f^j_{\vec{X}'(1)}, f^j_1$ is a little difficult to be represented.

## 4.3 Experiments

The test function and all experiment's conditions are the same as the subsection 3.3. Fig.6 shows the simulation results under $m = 1, 2$ respectively. The upper part shows the mean function value while the lower part shows the standard deviation of function value.

According to the simulation results, we can get some following conclusions.

- From comparison between Fig.4 and Fig.6, we can see the mean fuction value and the standard deviation of Fig.6 are smaller than those of Fig.4. It means RSG can increase the searching ability and robustness of GA.
- Comparing $m = 1$ and $m = 2$, we can see the mean fuction value and the standard deviation of $m = 2$ are smaller than those of $m = 1$. It means the increase of $m$ can make GA search for solutions more robustly.

From Fig.7 and Fig.5. we can see that the population concentrating rates of Fig.7 are slower than those of

Fig.5 while the concentrating rate of RGS under $m = 2$ is slower than that of RGS under $m = 1$.

## 5 Conclusion

In this paper, we focus on how the crossover and mutation work in GA by analyzing the variance of the bit's frequency and a new recombination method named RCBF is proposed. This method can make the population to distribute uniformly as large as possible without changing the bit's frequency string. It can increase the searching ability and robustness against uncertainty of GA, especially when the feasible solution space is very large. Based on RCBF, a new generation strategy named RGS is proposed where the $t+m$ th generation is determined not only by the $t+m-1$ th generation but also by generations from the $t+m-2$ th to the $t$ th. Some experiments have clarified that RGS increases the searching ability and roubustness of GA as well.

## References


[1] J. H. Holland, "Adaptation in Natural and Artificial Systems." Ann Arbor, MI: University of Michigan Press 1975.

[2] D. E. Goldberg, "Genetic Algorithms in Search, Optimization and Machine Learning", Addison-Wesley, Reading, MA 1989.

[3] Alex Rogers, Adam Prügel-Bennett, "Genetic Drift in Genetic Algorithm Selection Schemes", IEEE Trans. Evol. Comput., vol. 3, no. 4. pp.298-303, 1999.

[4] J.E. Baker, " Adaptive selection methods for genetic algorithms", in Proceeding of the First International Conference on Genetic Algorithm, Lawrence Erlbaum Associates, Hillsdale, NJ, pp. 101-111 1985.

[5] Z. Michalewicz, "Genetic Algorithm + Data Structures = Evolution Programs", second, extended edition, Springer 1994

[6] X. Yao and Y.Liu, "Fast evolution strategies," IEEE Trans. Evol. Comput. Vol. 3, no. 2. pp.82-102, 1999.

[7] Yiu-Wing Leung and Yuping Wang, "An Orthogonal Genetic Algorithm with Quantization for Global Numerical Optimization," IEEE Trans. Evol. Comput. Vol. 5, no. 1. pp.41-53, 2001.