Journal of Intelligent Learning Systems and Applications, 2015, 7, 37-41
Published Online May 2015 in SciRes. http://www.scirp.org/journal/jilsa
http://dx.doi.org/10.4236/jilsa.2015.72004

Iterated Function System-Based Crossover Operation for Real-Coded Genetic Algorithm

S. H. Ling

Centre for Health Technologies, University of Technology, Sydney, Australia
Email: steve.ling@uts.edu.au

Received 3 March 2015; accepted 13 April 2015; published 15 April 2015

Abstract

An iterated function system crossover (IFSX) operation for real-coded genetic algorithms (RCGAs) is presented in this paper. Iterated function system (IFS) is one type of fractals that maintains a similarity characteristic. By introducing the IFS into the crossover operation, the RCGA performs better searching solution with a faster convergence in a set of benchmark test functions.

Keywords

Genetic Algorithm, Iterated Function System, Crossover Operation

1. Introduction

Genetic algorithm (GA) [1] is a stochastic search algorithm for solving optimization problems. It can help find out the globally optimal solution over a domain. In general, three genetic operations affect the performance of the GA: selection, crossover and mutation. Selection operation selects the parents from the population with respect to some probability distribution and the fitness values. The crossover operation combines the information of the selected parents and generates the offspring. The mutation operation introduces changes to the offspring variables. Recently, different crossover operations for real-coded GA have been proposed to improve the efficiency. The unimodal normal distribution crossover (UNDX) was proposed [2] for multi-modal and highly epistatic functions. The blend crossover (BLX-α) [3] was reported showing good search ability for separable fitness functions. Average-bound crossover [4] was introduced to enhance the solution quality and solution stability. In this paper, a new crossover operation is presented.

Iterated function system (IFS) theory was proposed by Barnsley [5], which involved a specific fractal that enhances a self-similarity property. Based on the IFS, objects are dissected into pieces that are similar to the whole object. Taking advantage of the self-similarity property of IFS, an iterated function system crossover (IFSX) is proposed for real-coded GAs. It will be shown that the GA with IFSX will perform more efficiently

How to cite this paper: Ling, S.H. (2015) Iterated Function System-Based Crossover Operation for Real-Coded Genetic Algorithm. *Journal of Intelligent Learning Systems and Applications*, **7**, 37-41. http://dx.doi.org/10.4236/jilsa.2015.72004

and provide a faster convergence in a suite of benchmark test functions.

2. Iterated Functions System Crossover

The idea of using IFSX to reproduce offspring is shown in **Figure 1**. The procedure is as follows:

1) Select 2 parents $\boldsymbol{p}_1 = \begin{bmatrix} p_{11} & p_{12} & \cdots & p_{1n_{para}} \end{bmatrix}$ and $\boldsymbol{p}_2 = \begin{bmatrix} p_{21} & p_{22} & \cdots & p_{2n_{para}} \end{bmatrix}$ from the population, where n_{para} is the number of parameters.

2) Combine the information (genes) of $\boldsymbol{p}_1$ and $\boldsymbol{p}_2$ to form a vector $\boldsymbol{v}$ of complex elements given by

$$\begin{bmatrix} v_1 & v_2 & \cdots & v_{n_{para}} \end{bmatrix} = \boldsymbol{v} = \begin{bmatrix} p_{11} + jp_{21} & p_{12} + jp_{22} & \cdots & p_{1n_{para}} + jp_{2n_{para}} \end{bmatrix} \quad (1)$$

3) Based on the IFS theory [4], let

$$\hat{v}_{ij} = v_i + \lambda v_j \quad (2)$$

where $i, j = 1, 2, \cdots, n_{para}$, $\lambda \in (0,1]$ is a scaling factor. $\hat{v}_{ij}$ are the possible values generated by the IFS that exhibits a self-similarity property. For instance, in **Figure 1**, there are 3 values, v_1, v_2 and v_3. From (2), we have 9 values $\hat{v}_{ij}$, i, j = 1, 2, 3. This figure shows a fractal, and the patterns of $\hat{v}_{1j}, \hat{v}_{2j}$ and $\hat{v}_{3j}$, j = 1, 2, 3 are similar to the pattern of v_1, v_2 and v_3. In some cases, the value of $\hat{v}_{ij}$ may be out of the boundary. Then, the system will generate a random value (within the boundary) to replace it.

4) Randomly pick up n_{para} variables from $\hat{v}_{ij}$. If $\hat{V} = \{\hat{v}_{ij}\}$,

$$\text{then } Q = \{ q_1 \quad q_2 \quad \cdots \quad q_{n_{para}} \} \subset \hat{V} \quad (3)$$

For example, in **Figure 1**, a possible Q can be $\{ \hat{v}_{21} \quad \hat{v}_{23} \quad \hat{v}_{31} \}$.

5) Generate the offspring $\hat{\boldsymbol{P}}_1$ and $\hat{\boldsymbol{P}}_2$ as follows:

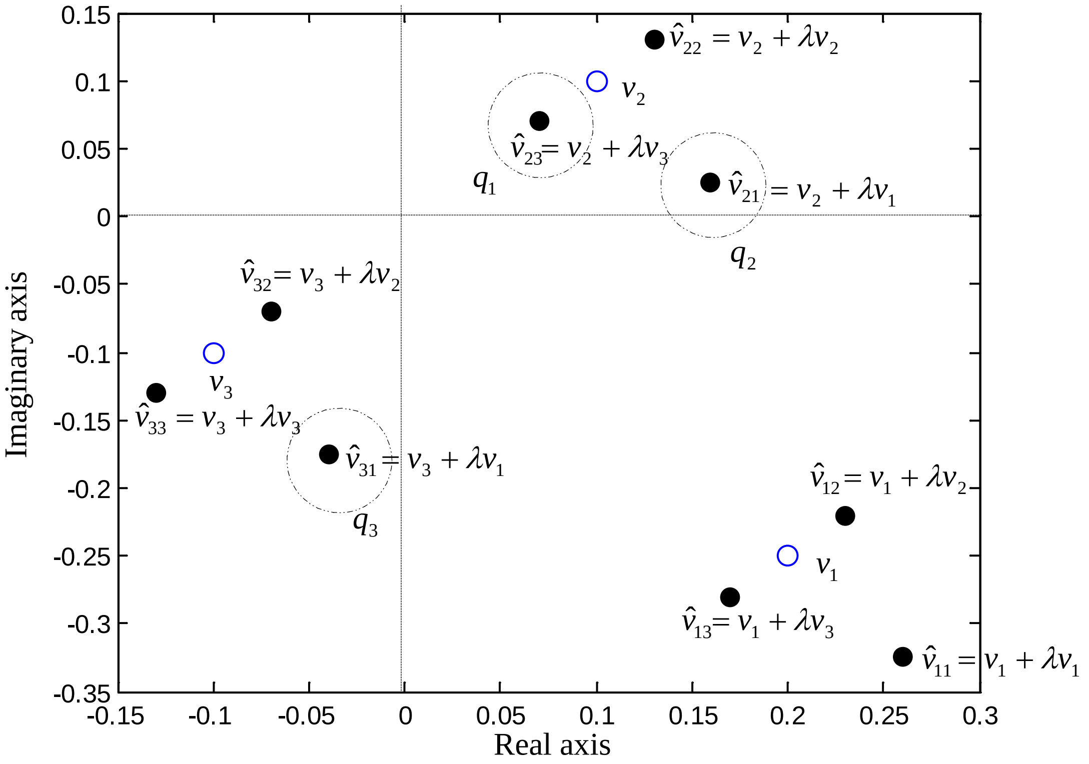


Figure 1. Idea of the proposed IFSX.

$$\hat{\boldsymbol{P}}_1 = \mathrm{Re}(Q) \tag{4}$$

$$\hat{\boldsymbol{P}}_2 = \mathrm{Im}(Q) \tag{5}$$

where Re(Q) and Im(Q) generate vectors formed by the real part and imaginary part of the elements of Q respectively.

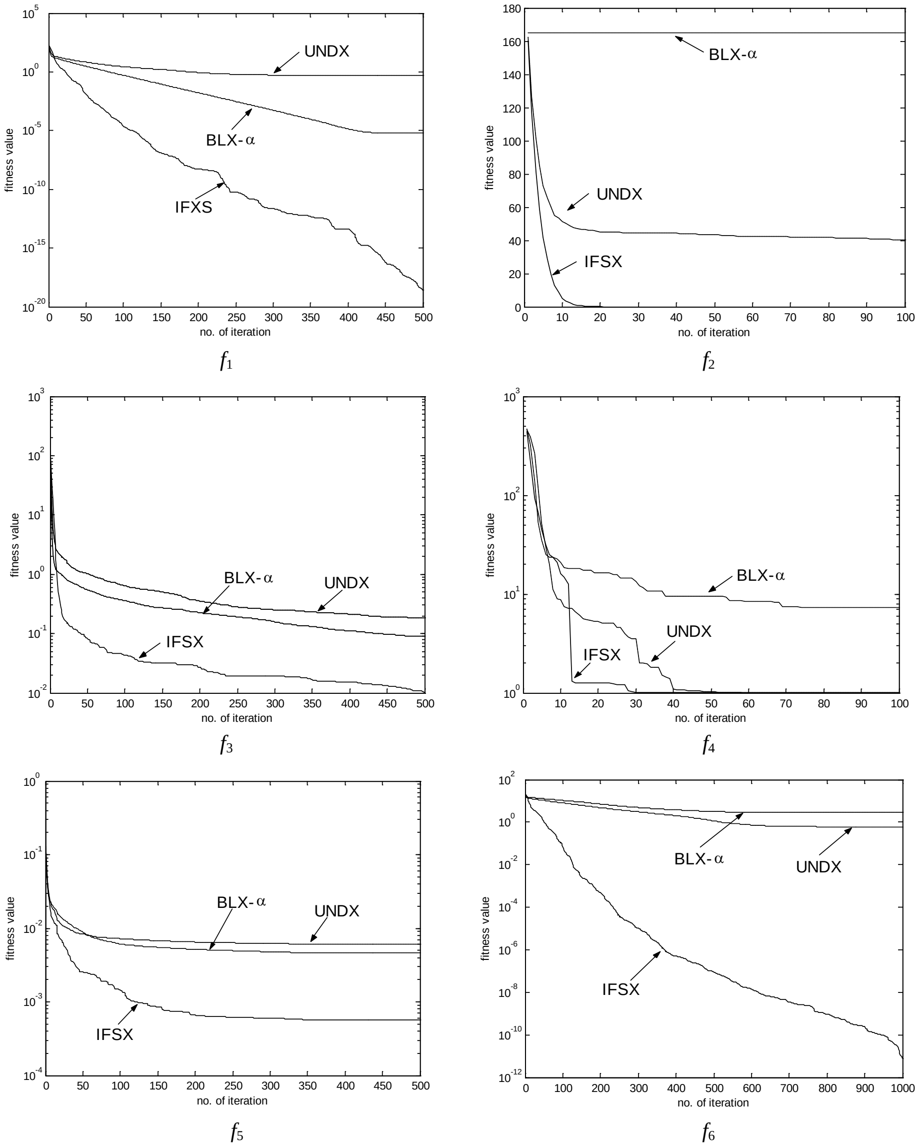
Crossover operation is mainly for exchanging information from the two selected parents. In traditional crossover operations (e.g. UNDX and BLX-α), the information is exchanging gene by gene. In the proposed IFSX, each offspring gene is affected by all other genes of the parents, which is a more "complete" crossover operation for information exchange. The IFSX crossover makes the GA operation performs better in terms of fitness value and convergence rate.

3. Simulation Results

The GA with the proposed IFSX goes through six test functions. The results are compared to those from GAs with UNDX and BLX-α. For each test function, the population size is 50 and all the simulation results are averaged ones out of 50 runs. The selection algorithm and the mutation operation are the roulette wheel selection [1] and the non-uniform mutation [1] respectively. The six test functions are listed in **Table 1**. f_1 is a sphere model which is smooth and symmetric. f_2 is a step function, which is a representative of flat surfaces. f_3 is a quartic function which is a simple unimodal function padded with noise. f_4 is a Shekel's foxholes function and f_5 is a Kowalik's function, which are multimodel functions with only a few local minima. f_6 is an Ackley's function which is a multimodel function with many local minima. The parameter λ of the IFSX are set at 0.005, 0.001, 0.001, 0.01, 0.005, 0.005 for f_1 to f_6 respectively, and the parameter of the BLX-α is set at 0.336 [3]. The simulation results obtained by the GA with the proposed IFSX, UNDX and BLX-α are shown in **Figure 2** and **Table 2**. It can be seen that the searching performance of the proposed IFSX is improved with faster convergence rate.

Table 1. Six benchmark test functions.

Test functions	Optimal point
$f_1(\boldsymbol{x}) = \sum_{i=1}^{30} x_i^2$, $-5.12 \le x_i \le 5.12$	$f_1(\boldsymbol{0}) = 0$
$f_2(\boldsymbol{x}) = \sum_{i=1}^{30} \left(\left\lfloor (x_i + 0.5)^2 \right\rfloor \right)$, $-0 \le x_i \le 5$	$f_2(\boldsymbol{0}) = 0$
$f_3(\boldsymbol{x}) = \sum_{i=1}^{30} i x_i^4 + random[0,1)$, $-1.28 \le x_i \le 1.28$	$f_3(\boldsymbol{0}) = 0$
$f_4(\boldsymbol{x}) = \left[\frac{1}{500} + \sum_{j=1}^{25} \frac{1}{j + \sum_{i=1}^{2} (x_i - a_{ij})^6} \right]^{-1}$, $-65.536 \le x_i \le 65.536$, $a_{ij} = \begin{pmatrix} -32 & -16 & 0 & 16 & 32 & -32 & \cdots & 0 & 16 & 32 \\ -32 & -32 & -32 & -32 & -32 & -16 & \cdots & 32 & 32 & 32 \end{pmatrix}$	$f_4([-32 \quad -32]) \approx 1$
$f_5(\boldsymbol{x}) = \sum_{i=1}^{11} \left[a_i - \frac{x_1 (b_i^2 + b_i x_2)}{b_i^2 + b_i x_3 + x_4} \right]^2$, $-5 \le x_i \le 5$, $\boldsymbol{a}$ = [0.1957 0.1947 0.1735 0.1600 0.0844 0.0627 0.0456 0.0342 0.0323 0.0235 0.0246], $\boldsymbol{b}$ = [4 2 1 0.5 0.25 0.167 0.125 0.1 0.0833 0.0714 0.0625]	$f_5([0.1928 \quad 0.1908 \quad 0.1231 \quad 0.1358])$ ≈ 0.0003075
$f_6(\boldsymbol{x}) = -20 \exp\left(-0.2 \sqrt{\frac{1}{30} \sum_{i=1}^{30} x_i^2} \right) - \exp\left(\frac{1}{30} \sum_{i=1}^{30} \cos 2\pi x_i \right) + 20 + e$, $-32 \le x_i \le 32$,	$f_6(\boldsymbol{0}) = 0$



f_1 f_2 f_3 f_4 f_5 f_6

Figure 2. Simulation results for f_1 to f_6 based on the proposed IFSX (solid line), UNDX (dashed line) and BLX-α (dotted line).

Table 2. Statistical results for f_1 to f_6.

		IFSX	UNDX	BLX-α
f_1	Ave.	2.6783e−19	4.9364e−1	6.0905e−6
	S.D.	9.8771e−19	7.6386e−1	5.2821e−6
f_2	Ave.	0	8.5	164.82
	S.D.	0	5.6973	16.241
f_3	Ave.	9.9721e−3	1.8537e−1	8.9198e−2
	S.D.	2.1561e−2	1.7033e−1	4.1623e−2
f_4	Ave.	0.99942	1.0023	7.2684
	S.D.	0.00688	0.02393	43.494
f_5	Ave.	5.6569e−4	6.1026e−3	4.6089e−3
	S.D.	7.9694e−4	1.0703e−2	6.8570e−3
f_6	Ave.	7.444e−12	2.8856	6.0939e−1
	S.D.	2.0494e−11	2.4163	6.6202e−1

4. Conclusion

In this paper, a new crossover of IFSX for real-coded GA has been proposed. Take the advantage of the iterated function system theory and integrate into crossover operation of real-code genetic algorithm, the solution quality of the searching is enhanced. A suite of benchmark test functions has been used to illustrate the merits of the IFSX.

References

[1] Michalewicz, Z. (1994) Genetic Algorithm + Data Structures = Evolution Programs. 2nd Edition, Springer, Berlin Heidelberg, New York. http://dx.doi.org/10.1007/978-3-662-07418-3

[2] Ono, I. and Kobayashi, S. (1997) A Real-Coded Genetic Algorithm for Function Optimization Using Unimodal Normal Distribution Crossover. *Proceedings of the Seventh International Conference on Genetic Algorithms*, USA, 19-23 July 1997, 246-253.

[3] Eshelman, L.J. and Schaffer, J.D. (1993) Real-Coded Genetic Algorithms and Interval-Schemata. *Foundations of Genetic Algorithms*, **2**, 187-202. http://dx.doi.org/10.1016/B978-0-08-094832-4.50018-0

[4] Ling, S.H. and Leung, F.H.F. (2007) An Improved Genetic Algorithm with Average-Bound Crossover and Wavelet Mutation Operations. *Soft Computing*, **11**, 7-31. http://dx.doi.org/10.1007/s00500-006-0049-7

[5] Barnsley, M.F. (1993) Fractals Everywhere. Academic Press, Cambridge.