Maintaining the Diversity of Solutions by Non-Geometric Binary Crossover: A Worst One-Max Solver Competition Case Study

Hisao Ishibuchi, Noritaka Tsukamoto, and Yusuke Nojima
Graduate School of Engineering, Osaka Prefecture University
1-1 Gakuen-cho, Naka-ku, Sakai, Osaka 599-8531, Japan
Phone: +81-72-254-9350
hisaoi@cs.osakafu-u.ac.jp, nori@ci.cs.osakafu-u.ac.jp, nojima@cs.osakafu-u.ac.jp

ABSTRACT

The worst one-max solver competition task in GECCO 2007 was to develop a one-max solver that can find the optimal solution of the 15-bit one-max problem as late as possible within 1000 generations. There are two conflicting issues in developing such a one-max solver. One is to slow down the evolution of solutions toward the optimal solution (i.e., not to find the optimal solution in early generations). The other is to find the optimal solution in a very late generation. In this paper, we examine the effect of using a non-geometric binary crossover operator through computational experiments on the worst one-max solver competition task.

Categories and Subject Descriptors

I.2.8 [**Artificial Intelligence**]: Problem Solving, Control Methods, and Search – *Heuristic Methods.*

General Terms

Algorithms.

Keywords

Genetic algorithms, binary crossover, non-geometric crossover, diversity maintenance.

1. INTRODUCTION

Recently the concept of geometric crossover was proposed by Moraglio and Poli [2] to analyze crossover operators in terms of the distances between an offspring and its parents. Roughly speaking, a crossover operator is referred to as being geometric when the following relation always holds between an offspring C and its two parents P1 and P2:

$$Distance(\text{C, P1}) + Distance(\text{C, P2}) = Distance(\text{P1, P2}), \qquad (1)$$

where *Distance*(A, B) denotes the distance between A and B.

Traditional mask-based crossover operators for binary strings (e.g., one-point, two-point and uniform) are geometric because the relation in (1) always holds for the Hamming distance. This means that such a crossover operator always generates an offspring in the segment between its parents under the Hamming distance. On the other hand, non-geometric crossover operators are often used for real number strings. Such a crossover operator generates an offspring C satisfying the following relation:

$$Distance(\text{C, P1}) + Distance(\text{C, P2}) > Distance(\text{P1, P2}), \qquad (2)$$

where the Euclidean distance is used to measure the distance.

The use of non-geometric binary crossover was proposed for multiobjective optimization in Ishibuchi et al. [1]. They clearly demonstrated that the use of non-geometric binary crossover in NSGA-II increased the diversity of solutions. In this paper, we examine its use for the worst one-max solver competition task.

2. NON-GEOMETRIC CROSSOVER

In order to generate an offspring outside the segment between its two parents, a non-geometric binary crossover operator was proposed in Ishibuchi et al. [1]. Let **x** and **y** be two binary strings of length n. We denote them as $\mathbf{x}=x_1 x_2 ... x_n$ and $\mathbf{y}=y_1 y_2 ... y_n$ (x_i=0/1 and y_i=0/1 for $i=1,2,...,n$). The non-geometric binary crossover operator in [1] generates an offspring $\mathbf{z}=z_1 z_2 ... z_n$ satisfying the following relation:

$$Distance(\mathbf{x}, \mathbf{z}) + Distance(\mathbf{y}, \mathbf{z}) > Distance(\mathbf{x}, \mathbf{y}), \qquad (3)$$

where the distance is measured by the Hamming distance.

The basic idea of the non-geometric binary crossover operator [1] is to generate an offspring from one parent in the opposite side of the other parent. First one parent is chosen as a primary parent (say **x**). The choice of a primary parent can be random or based on the fitness of each parent. The other parent (say **y**) is used as a secondary parent. Then an offspring **z** is generated from the primary parent **x** and the secondary parent **y** as follows:

[Non-geometric binary crossover for the two parents **x** and **y**]

Step 1: Let $z_i := x_i$ for $i=1,2,...,n$ (i.e., **z** is a copy of the primary parent **x**).

Step 2: Let $z_i := (1-z_i)$ with a probability P_{BF} when $x_i = y_i$ where P_{BF} is a prespecified bit-flip probability.

In this crossover operator, the bit-flip operation is applied to x_i of the primary parent with a prespecified probability P_{BF} only when $x_i = y_i$ (i.e., only when the values are the same between the two parents). On the other hand, x_i is always inherited to the offspring with no changes when $x_i \neq y_i$.

GECCO'08, July 12-16, 2008, Atlanta, Georgia, USA.
ACM 978-1-60558-130-9/08/07.

3. COMPETITION TASK RESULTS

The worst one-max solver competition task was to design an evolutionary algorithm that can find the optimal solution of the 15-bit one-max problem as late as possible in 1000 generations. Linear changes of parameter values depending on the number of generations were allowed. Other explicit usage of the information on the number of generations was not allowed in the competition. Among 500 independent runs from random initial populations, it was requested to find the optimal solution in more than 475 runs (i.e., the success rate should be larger than 95%).

There are two issues in the design of such a worst one-max solver. One is to prevent solutions from evolving toward the optimal solution until the final stage of evolution. The other is to almost always find the optimal solution in the final stage of evolution. We examined the performance of a simple generational genetic algorithm with no elite solutions (i.e., a 100% generation gap). Two parents were selected by binary tournament selection. One of the two parents was randomly chosen as a primary parent in the non-geometric binary crossover operator. The number of one's in each binary string of length 15 was used as its fitness value.

Our trick for the worst one-max solver design in this paper is to linearly decrease the bit-flip probability P_{BF} in the non-geometric binary crossover operator of our genetic algorithm as

$$P_{BF} = 1 - T/1000\,, \tag{4}$$

where T is the number of generations ($T = 1, 2, ..., 1000$).

In our genetic algorithm, the standard bit-flip mutation was applied to each bit value of strings with the probability P_M. On the other hand, the non-geometric binary crossover operator was applied to each string with the crossover probability P_C. We did not use any geometric crossover operators. We examined all the 6×5 combinations of the following values of P_C and P_M:

P_C = 0.5, 0.6, 0.7, 0.8, 0.9, 1.0,
P_M = 0.001, 0.002, 0.004, 0.008, 0.010.

Our genetic algorithm with each combination of P_C and P_M was applied to the 15-bit one-max problem 500 times. The population size was specified as 20. The optimal solution was found in almost all runs. All combinations of P_C and P_M satisfied the requirement of the competition: 95% success rate.

The average number of generations required to find the optimal solution of the 15-bit one-max problem is shown in Fig. 1. For example, the optimal solution was obtained around the 900th generation on average in Fig. 1 when the crossover probability P_C was 1.0. A few unsuccessful runs were not included in the calculation of the average number of generations in Fig. 1.

In Fig. 2, we show the average values of the worst, average, and best fitness at each generation over 500 runs of our genetic algorithm with P_C = 1.0 (non-geometric binary crossover) and P_M = 0.01. In Fig. 2, the use of the non-geometric binary crossover operator clearly slowed down the evolution of solutions toward the optimal solution in the first 900 generations.

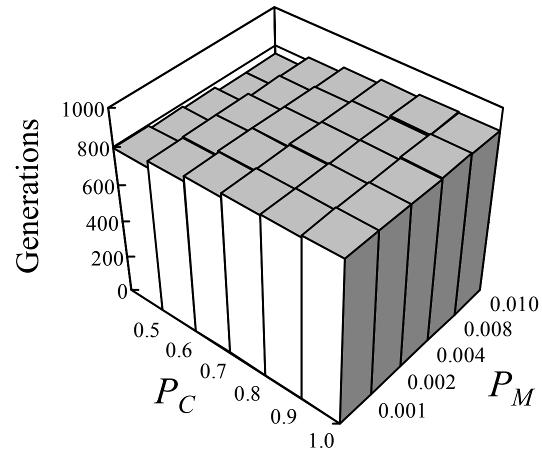

Figure 1. Average number of generations required to find the optimal solution where the non-geometric crossover operator was used with the probability P_C. (500 runs)

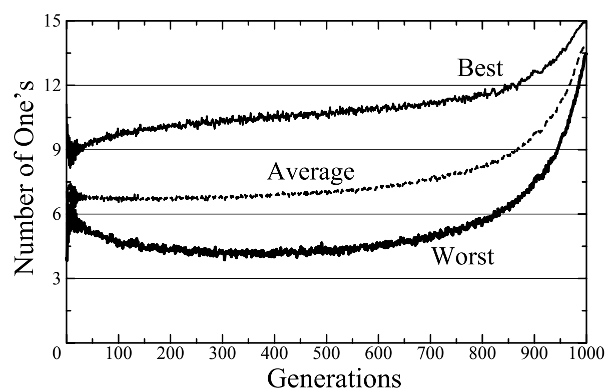

Figure 2. Average values of the worst, average and best fitness at each generation over 500 runs of our genetic algorithm with the non-geometric binary crossover operator.

4. CONCLUDING REMARKS

In this paper, we examined the effect of a non-geometric binary crossover operator on the search behavior of genetic algorithms through computational experiments on the worst one-max solver competition task in GECCO 2007.

5. REFERENCES

[1] Ishibuchi, H., Nojima, Y., Tsukamoto, N., and Ohara, K. Effects of the use of non-geometric binary crossover on evolutionary multiobjective optimization. *Proc. GECCO 2007*, pp. 829-836.

[2] Moraglio, A., and Poli, P. Topological interpretation of crossover. *Proc. GECCO 2004*, pp. 1377-1388.