# A Preliminary Study of Crowding with Biased Crossover

Shigeyoshi Tsutsui
Hannan University
5-4-33, Amami-higashi, Matsubara
Osaka 580-8502, Japan
tsutsui@hannan-u.ac.jp

Noriyuki Fujimoto
Osaka Prefecture University
1-1 Gakuen-cho, Nakaku, Sakai
Osaka 599-8531, Japan
fujimoto@mi.s.osakafu-u.ac.jp

## ABSTRACT
This paper proposes a novel crowding method, which is called "Crowding with Biased Crossover (CBX)". The Biased crossover operator begins with two parents. Then two offspring individuals are created, each offspring taking more characteristics from one of the two parents. This is an easy method to perform replacement between parents and offspring individuals. Experimental results showed that CBX is very effective in finding both single global solutions and multiple solutions (niching).

## Categories and Subject Descriptors
I.2.8 [Artificial Intelligence]: Problem Solving, Control Methods, and Search Heuristic Methods

## General Terms
Algorithms

## Keywords
Crowding, Genetic Algorithms (GAs), Niche, Niching

## 1. INTRODUCTION
Crowding methods constitute an important research area in genetic and evolutionary computation. There are two main objectives of crowding methods: (1) one is to prevent premature convergence of a population by preserving the population diversity, and obtain one global optimal solution; (2) the other is to converge the population to multiple, highly fit, and significantly different solutions (niching) [2].

In this paper, we propose a novel crowding method, which we call "Crowding with Biased Crossover (CBX)" and show promising results with CBX using several test problems. In the literature of crowding methods, the main efforts are focused on how replacement is performed between parents individuals and offspring individuals using similarity between them as a replacement criteria. CBX does not use the similarity as a criteria for replacement. Instead, we use a "biased crossover" for crossover operators. The biased crossover operator generates offspring individuals which are each similar to one of two parent individuals. The degree of the similarity between the parents individuals and the offspring individuals is controlled by a parameter. By choosing the value of the parameter, CBX can maintain population diversity to obtain one global optimal solution, or converge the population to multiple different solutions (niching).


*GECCO'13 Companion,* July 6–10, 2013, Amsterdam, The Netherlands.
ACM 978-1-4503-1964-5/13/07.

## 2. A BRIEF REVIEW OF CROWDING METHODS
Here, we review typical crowding methods. Crowding consists of two main phases: pairing and replacement. In the pairing phase, offspring individuals are paired with individuals in the current population according to a similarity metric. In the replacement phase, a decision is made for each pair of individuals as to which of them will remain in the population [4].

**(1) Crowding factor model:** The main purpose of the crowding factor model by De Jong [1] is to maintain population diversity. In the crowding factor model, replacement for each offspring produced is considered individually. For each such individual, a sample of *CF* (Crowding Factor) individuals are drawn from the population and searched for the most similar individual to the offspring in question. The most similar individual from the small sample is then directly replaced in the population by the offspring, without regard for fitness.

**(2) Deterministic Crowding (DC):** Since offspring are obtained by recombination of their parents, parent individuals and offspring individuals have a certain degree of similarity. Deterministic Crowding (DC) uses this feature as shown in Figure 1 [3].

1. Select two parents, $p_1$ and $p_2$, randomly, without replacement
2. Cross them, yielding $c_1$ and $c_2$
3. If $|p_1, c_1| + |p_2, c_2| \leqq |p_1, c_2| + |p_2, c_1|$
   - If $f(c_1) > f(p_1)$, replace $p_1$ with $c_1$
   - If $f(c_2) > f(p_2)$, replace $p_2$ with $c_2$

   Else
   - If $f(c_2) > f(p_1)$, replace $p_1$ with $c_2$
   - If $f(c_1) > f(p_2)$, replace $p_2$ with $c_1$

**Figure 1. Deterministic crowding methods.**

**(3) Probabilistic Crowding (PC), Boltzmann Crowding (BC):** Unlike DC, PC uses a non-deterministic rule to establish the winner of a competition between parent *p* and child *c*. BC is based on the well-known Simulated Annealing method, implemented with the Boltzmann acceptance rule [2].

## 3. CROWDING WITH BIASED CROSS-OVER
As we saw in Section 2, in usual crowding methods, offspring individuals are generated using usual crossover operators in the pairing phase and then the similarity between parents and offspring individuals are measured in the replacement phase. In Crowding with Biased Crossover (CBX), CBX does not use the similarity measure in the replacement phase. Instead, a "biased crossover (BX)" in the pairing phase is used. The BX generates two offspring individuals each which is similar to one of the two parent individuals.

Although BX is not restricted to 2-point crossover, here we explain the BX using 2-point crossover for simplicity. Let $l_c$ be the length between cut-point cut1 and cut2. Since $l_c$ distributes in $[1, n-1]$ uniformly, the average value of $l_c$, $E(l_c)$, is $n/2$, where $n$ is the string length (or problem size).

In BX, we sample two cut-points so that $E(l_c)$ is bigger than $n/2$. If we choose two cut-points so that $E(l_c)$ is closer to $n$, then both offspring individuals $c_1$ and $c_2$ are more similar to parents $p_1$ and $p_2$, respectively. To control the similarity, BX introduces a parameter $\lambda$ ($0.5 \le \lambda < 1$) which controls the similarity by sampling $l_c$ as $E(l_c) = n \times \lambda$. For probability density function (p.d.f.) of $l_c$, we used the following p.d.f. function which was used in our previous study on *c*AS (cunning Ant System, see reference [6] for details).

$$p(l_c) = \frac{\lambda}{n(1-\lambda)} \left( \frac{l_c}{n} \right)^{\frac{2\lambda-1}{1-\lambda}}. \quad (2)$$

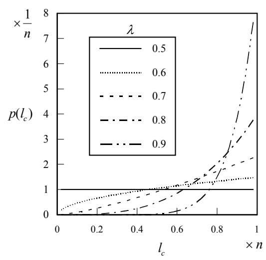


Figure 3 shows the BX and Figure 4 shows the CBX algorithm. In the BX, if we set $\lambda$=0.5, BX is identical to 2-point, since $p(l_c)=1/n$ and $E(l_c)=n/2$.

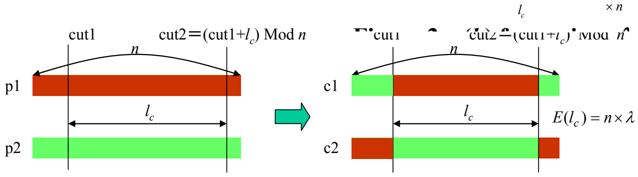


**Figure 3. Biased Crossover (BX).**

1. Set $\lambda$ (e.g. $\lambda$=0.8)
2. Select two parents, $p_1$ and $p_2$, without replacement
3. Sample cut1 randomly in $[0,n-1]$
4. Sample $l_c$ according to Eq. (2)
5. Set cut as cut2=(cut1+ $l_c$) Mod $n$
6. Cross $p_1$ and $p_2$, yielding $c_1$ and $c_2$
   - If $f(c_1)>f(p_1)$, replace $p_1$ with $c_1$
   - If $f(c_2)>f(p_2)$, replace $p_2$ with $c_2$

**Figure 4. Crowding with Biased Crossover (CBX).**

# 4. EXPERIMENTS

In this section, we perform two types of experiments with CBX; (1) To solve the traveling salesman problems (TSPs) to obtain a single solution, (2) To solve multimodal problems to obtain multiple solutions.

## 4.1 Solving TSPs with CBX

In this experiment, we apply CBX to small size TSP instances that are below 100 cities. Here, we use no heuristics. Order Crossover (OX) is used for the base of CBX. Population size is set to $2 \times n$. We run the algorithm until number of evaluations reaches $10{,}000 \times n$. 10 runs were performed for each instance. Table 1 summarized the results. Here, *Error* is the average excess rate from optimum length over 10 runs. From these results, we can see that by choosing a value of $\lambda$ larger than 0.5 ($\lambda$=0.7 or 0.8), we can obtain good quality solutions.

**Table 1. Results of CBX on the small TSP instances.**

| Instances | *Error* | | | | | |
|---|---|---|---|---|---|---|
| | Simple GA | CBX | | | | |
| | | $\lambda$=0.5 | $\lambda$=0.6 | $\lambda$=0.7 | $\lambda$=0.8 | $\lambda$=0.9 |
| gr48 | 2.767% | 0.951% | 0.822% | 0.727% | **0.182%** | 6.449% |
| eil51 | 4.225% | 2.324% | 2.394% | 1.620% | **1.573%** | 16.033% |
| berlin52 | 5.231% | 3.467% | 2.468% | 0.617% | **0.000%** | 12.151% |
| pr76 | 12.473% | 3.402% | 2.118% | **1.750%** | 12.884% | 69.458% |
| kroA100 | 47.230% | 3.605% | 2.567% | **1.897%** | 104.584% | 144.186% |

## 4.2 Niching for multimodal problems

Now we show how CBX converges the population to multiple, highly fit, and significantly different solutions (niching). We use the following two functions [5].

$$f_1(x) = \sin^6(5\pi x), \quad (3)$$

$$f_2(x) = e^{-2(\ln 2)\left(\frac{x-0.1}{0.8}\right)^2} \sin^6(5\pi x). \quad (4)$$

We encoded $x$ in the range [0, 1] with 30-bit binary string. Population size was 100. As shown in Figure 4, CBX found multiple solutions with larger value of $\lambda$ than 0.5.

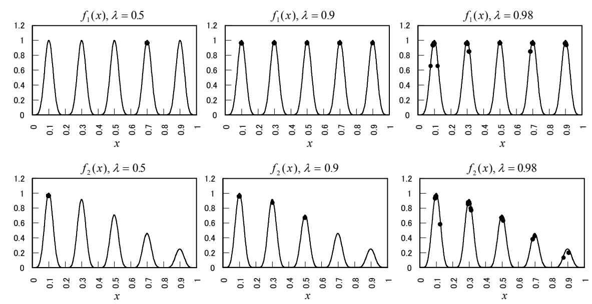


**Figure 4. Niching with CBX.**

# 5. CONCLUSION

In this paper, we proposed a novel crowding method, Crowding with Biased Crossover (CBX). Experimental results showed that CBX is very effective in finding both single global solutions and multiple solutions (niching). We need to evaluate CBX on various problems and this remains for future work.

# 6. REFERENCES


[1] K. A. De Jong. An analysis of the behavior of a class of genetic adaptive systems. PhD thesis, Dept. of Computer pand Comm. Sciences, Univ. of Michigan, 1975.

[2] O. J. Mengshoel and D. E. Goldberg. The crowding approach to niching in genetic algorithms. Evolutionary Computation, 16(3), pp. 315–354, 2008.

[3] S. W. Mahfoud. Niching methods for genetic Algorithms. PhD thesis, Department of General Engineering, University of Illinois at Urbana-Champaign, Urbana, IL, 1995.

[4] S. F. Galan and O. J. Mengshoel. Generalized crowding for genetic algorithms, GECCO 2010, pp. 775-782, 2010.

[5] D. E. Goldberg and J. Richardson. Genetic algorithms with sharing for multimodal function optimization, Proceedings of the 2nd Inter. Conf. on Genetic Algorithms and their application, pp. 41-49, 1987.

[6] S. Tsutsui. *c*AS: Ant Colony Optimization with Cunning Ants Proc. of the PPSN IX, pp. 162-171, 2006.