# Investigation of Mutation Operators for the Bayesian Optimization Algorithm

Eric M. Heien, Noriyuki Fujimoto
Graduate School of Information Science and Technology, Osaka University
Toyonaka, Osaka 560-8531, Japan
{e-heien, fujimoto}@ist.osaka-u.ac.jp

Tomoyuki Hiroyasu
Department of Knowledge Engineering
Doshisha University
Kyo-tanabe, Kyoto, 610-0321, Japan
tomo@is.doshisha.ac.jp

## Categories and Subject Descriptors

G.1.6 [**Optimization**]: Global optimization; G.3 [**Probability and Statistics**]: Probabilistic algorithms (including Monte Carlo); I.2.8 [**Problem Solving, Control Methods, and Search**]: Heuristic methods

## General Terms

Performance

## Keywords

Bayesian optimization algorithm, mutation, probabilistic model building genetic algorithm

## ABSTRACT

A Bayesian network is a probabilistic graphical model, consisting of a directed acyclic graph whose nodes correspond to variables $X = (X_1, X_2, \ldots, X_\ell)$ to be modeled and whose edges correspond to conditional dependencies $p(X_i|\Pi_i)$ where $\Pi_i$ is the set of parents of $X_i$. The Bayesian optimization algorithm (BOA) is a probabilistic model building genetic algorithm that searches for optimal problem solutions by constructing a Bayesian network for a given training dataset.

Although BOA is effective at finding solutions for optimization problems, small population sizes in a model can result in premature convergence to a sub-optimal solution. One way of avoiding premature convergence is to increase population diversity with a mutation operator. In our experiments, we compare several mutation operators for use with BOA. We examine in detail the probabilistic model utilizing (PMU) bit flipping mutation operator. We compare the effectiveness of the PMU operator with standard BOA, self-adaptive evolution and local search of substructural neighborhoods.

The pseudocode for generating a new individual with PMU bitwise mutation given a Bayesian network is as follows:

1. Consider the first ungenerated variable $X_i$ based on the ancestral reverse ordering of variables in the Bayesian network.
2. Generate a value for variable $X_i$ based on the Bayesian network probability $p(X_i|\Pi_i)$.
3. With probability $p_m$, mutate the variable.
4. Repeat steps 1-3 for all remaining variables in ancestral reverse order.

**Table 1: Optimal mutation rates**

| Function | Maximize success | Minimize population |
|---|---|---|
| OneMax | $1/\ell \le p_m \le 3/\ell$ | $1/\ell \le p_m \le 3/\ell$ |
| 3-Deceptive | $3/\ell \le p_m \le 6/\ell$ | $p_m = 3/\ell$ |
| 5-Trap | $2/\ell \le p_m \le 4/\ell$ | $p_m = 3/\ell$ |
| 6-Bipolar | $1/\ell \le p_m \le 2/\ell$ | $1/\ell \le p_m \le 2/\ell$ |

**Table 2: Growth of fitness function evaluations**

| Function | $p_m = 0$ | $p_m = 1/\ell$ | $p_m = 2/\ell$ | $p_m = 3/\ell$ |
|---|---|---|---|---|
| OneMax | $O(\ell^{1.28})$ | $O(\ell^{1.26})$ | $O(\ell^{1.21})$ | $O(\ell^{1.07})$ |
| 3-Deceptive | $O(\ell^{1.84})$ | $O(\ell^{1.77})$ | $O(\ell^{1.81})$ | $O(\ell^{2.15})$ |
| 5-Trap | $O(\ell^{1.83})$ | $O(\ell^{1.96})$ | $O(\ell^{2.01})$ | $O(\ell^{1.99})$ |
| 6-Bipolar | $O(\ell^{1.49})$ | $O(\ell^{1.65})$ | $O(\ell^{1.77})$ | $O(\ell^{1.76})$ |

Four functions were used to test the PMU bitwise mutation operator: OneMax, 5-trap, 3-deceptive and 6-bipolar. Experiments were conducted to determine the effect of the mutation operator on the success rate for a given population size, the minimum required population size and total number of fitness function evaluations. The results for experiments of maximizing success rates and minimizing population are shown in Table 1. The results for experiments of total fitness function evaluations are shown in Table 2.

Based on the experimental results, we found that this type of mutation significantly increases the success rate of finding a function's optimum using a much smaller minimum population. This comes at a cost of more fitness function evaluations, though this cost appears to grow at a rate comparable to standard BOA. We compared PMU bitwise mutation to other BOA mutation operators and found that it tends to outperform self adaptive evolution for the 3-deceptive function, but underperform compared to substructural hill climbing for the 5-trap function.

The PMU bitwise mutation operator seems most suitable for functions where the optimal population size is unknown. The results show that using this mutation operator in BOA with proper mutation rates significantly increases the success rate and reduces the minimum required population size, while slightly increasing the number of fitness evaluations needed to find an optimal solution.

Full paper available at: http://www-hagi.ist.osaka-u.ac.jp/~e-heien/mut_boa_2007.pdf

*GECCO'07* July 7-11, 2007, London, England, United Kingdom
ACM 978-1-59593-697-4/07/0007.