The K-Bit-Swap: A New Genetic Algorithm Operator

Aram R. Ter-Sarkisov
SEAT, Massey University
New Zealand

Stephen R. Marsland
SEAT, Massey University
New Zealand

Barbara R. Holland
IFS, Massey University
New Zealand

ABSTRACT

Genetic algorithms (GA) mostly commonly use three main operators: selection, crossover and mutation, although many others have been proposed in the literature. This article introduces a new operator, k-bit-swap, which swaps bits between two strings without preserving the location of those bits, changing their order of bits in the string. It can be considered as a form of crossover. We investigate the effects of this operator and demonstrate that its use improves the speed and performance on several well-known problems.

Categories and Subject Descriptors: I.2.8 [Artificial Intelligence]: Problem solving, Control methods and search–*Heuristic methods*

General Terms: Algorithms, Performance, Verification

Keywords: Genetic algorithms, genetic operators, k-bit-swap operator, linear regression

1. THE K-BIT-SWAP OPERATOR

The k-bit-swap operator is a form of crossover, since information is exchanged between two chromosomes rather than within one; it most closely reflects uniform crossover [2], except that the location in the offspring string that a bit ends up at does not match its location in the parent string. Figure 1 shows the comparison. The method is related to transposition, a mechanism whereby genetic data is transferred between chromosomes [1]. The steps involved in k-bit-swap at each iteration are:

1. select two parent chromosomes for mating
2. repeat k times:
   (a) pick a bit location in each parent string
   (b) swap the values between those two bits

To investigate the performance we performed a set of experiments with and without k-bit-swap on nine different optimisation problems and tested 320 parameter settings (of population size, crossover, mutation, and k-bit-swap rates, etc., as summarised in Table 2) for each problem. Each experiment was run 100 times from initially random starting populations. The functions used have unique global optima, and a neighbourhood of the global optimum was chosen as representing success. Benchmarks used were the highest percentage of success runs with the best parameters (SR) and the average generation of this (AvGen).

To investigate how the variation in the success rate is explained by the independent variables (x_1: population size, x_2: crossover rate, x_3: k-bit-swap, x_4: mutation and x_5: elitism) we use a multiple linear regression of the form (where y is the response variable of success rate, SR):

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_3 + \beta_4 x_4 + \beta_5 x_5 + \epsilon$$

We measure the coefficient of determination, R^2, the amount of variation in the response variable explained by the explanatory variables. The results are shown in Table 3 for the optimal parameter values of each function being used. For most values the coefficient is quite low, meaning that these parameters are not sufficient to explain the success rate, with higher correlated with zero mutation or k-bit-swap.

In seven out of the nine functions that we investigated (all except the Travelling Salesman Problem) adding k-bit-swap improved the average generation that the vicinity of the global optimum was reached. It also improved the success rate for many of the functions, but again, not TSP. TSP is the only function that uses a non-binary string representation, and where the crossover operator is not the common one. In four of the seven functions (the first four in Table 1) k-bit-swap can be used as a replacement for both mutation and crossover and produce a high success rate. Indeed, for the rather lengthy (800 bits long) version of the Royal Roads function that we used it is the only operator that achieves very good results.

Table 1: Highest success rate (for best parameter setting) with and without k-bit-swap

Function	Success rate		AvGen	
	kbs	no kbs	kbs	no kbs
Rosenbrock	**1**	0.85	**142**	413
Ackley	1	1	**61**	80
Rastrigin	**1**	0.84	**53**	92
Royal Roads	**1**	N/A	**83**	N/A
Four Peaks	0.92	**0.95**	**386**	457
TSP (trivial)	0.04	**0.34**	376	**344**
TSP (US Cities)	N/A	**0.06**	N/A	**483**
2D approx (trivial)	1	1	**438**	462
2D approx (random)	**0.90**	0.88	**461**	471

Note: better results are highlighted in bold

2. REFERENCES

[1] A. Simoes and E. Costa. Transposition: A Biologically Inspired Mechanism To Use With Genetic Algorithms. In *Proceedings of the Fourth International Conference on Neural Networks and Genetic Algorithms*. Springer-Verlag, 1999.

[2] Gilbert Syswerda. Uniform crossover in genetic algorithms. In *Proceedings of the International Conference on Genetic Aglorithms*. Morgan Kaufmann, 1989.

GECCO'10, July 7–11, 2010, Portland, Oregon, USA.
ACM 978-1-4503-0072-8/10/07.

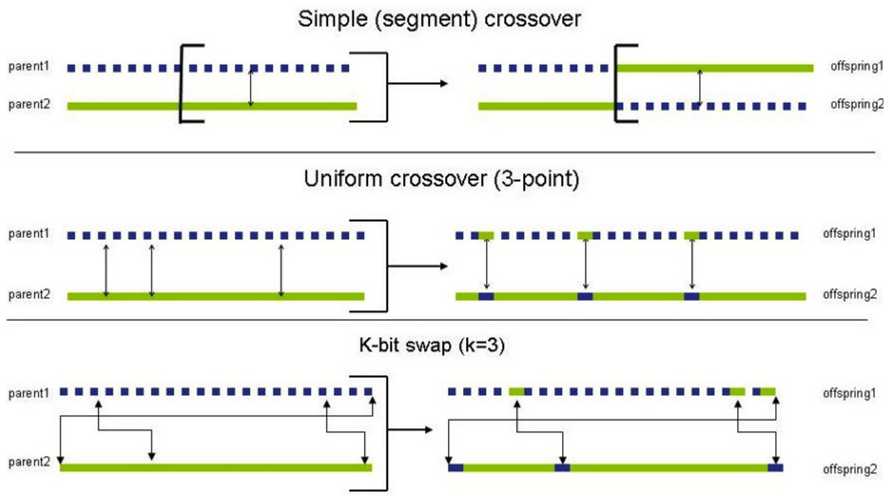

Figure 1: Comparison of k-bit-swap to Simple crossover and Uniform crossover

Table 2: Parameter Settings for the problem set

	PopSize	**Crossover rate**	**k-bit-swap**	**Mutation rate**	**ChromLength**	**Elitism**
Rosenbrock	50,100	0,simple,1,10,20	0,1,10,20	0,.025,.25,.5	40	1,50%
Ackley	50,100	0,simple,1,10,20	0,1,10,20	0,.025,.25,.5	40	1,50%
Rastrigin	50,100	0,simple,1,10,20	0,1,10,20	0,.025,.25,.5	40	1,50%
Royal roads	50,100	0,simple,1,100,400	0,1,100,400	0,.00125,.125,.5	800	none
Four peaks	50,100	0,simple,1,25,50	0,1,25,50	0,.01,.02,.05	100	1,50%
TSP (trivial)	50,100	0,simple,1,2,3	0,1,2,3	0,.038,.076,.115	26	1,50%
TSP (US cities)	50,100	0,simple,1,2,3	0,1,2,3	0,.02,.041,.0625	48	1,50%
2D Approximation	50,100	0,simple,1,5,10	0,1,5,10	0,.0125,.025,.05	160	1,50%

Note: PopSize is population size. Simple crossover is a single-point segment crossover. Other crossovers are uniform. Values for uniform crossover and k-bit-swap are the number of bits selected for swap. The values for elitism denote the number of chromosomes kept for the next generation. Fitness-proportional sigmoid selection is used for all functions.

Table 3: Regression Analysis

Function	**Overall**	**with k-bit-swap**	**without k-bit-swap**	**with crossover**	**without crossover**	**with mutation**	**without mutation**
	R^2	R^2	R^2	R^2	R^2	R^2	R^2
Rosenbrock	0.4439	0.3813	0.3574	0.4067	0.5835	0.3648	0.8862
Ackley	0.5271	0.7258	0.1004	0.5403	0.4935	0.5282	0.6048
Rastrigin	0.5584	0.6882	0.1301	0.5651	0.5387	0.5094	0.7690
Royal Roads	0.3878	0.4741	NaN	0.3894	0.3814	0.3048	0.6362
Four Peaks	0.4644	0.4192	0.6338	0.4809	0.5138	0.5364	0.2430
TSP(circle)	0.2547	0.1376	0.3917	0.2460	0.3033	0.3526	0.1178
TSP(US Capital cities)	0.0695	NaN	0.1592	0.0637	0.1226	0.0905	NaN
2D Approximation (trivial)	0.2655	0.4000	0.5284	0.2767	0.2660	0.6223	0.7349
2D Approximation (random)	0.3989	0.5295	0.5121	0.4178	0.3519	0.6389	0.6415