Uniform Crossover Revisited: Maximum Disruption in Real-Coded GAs

Stephen Drake

COGS, University of Sussex
Falmer, Brighton, BN1 9QH, UK

Abstract. A detailed comparison is presented of five well-known crossover operators on a range of continuous optimization tasks to test longheld beliefs concerning disruption, with respect to real-coded GAs. It is suggested that, contrary to traditional assumptions, disruption is useful.

1 Introduction

Five crossover operators were compared on a wide variety of continuous optimization tasks. The operators compared were traditional one-point (1PX), two-point (2PX), and uniform (UX, $P_0 = 0.5$)[6] crossover operators, together with two operators specifically designed for use with real-coded GAs: linear (LX) [7] and blend (BLX-0.5) [2].

2 Experiments

The comparison testbed included 'traditional' test functions, two versions of a real-world problem (aircraft wing-box optimization [5]) and an approximation to a real-world problem [4]; see [1] for details.

A geographically distributed GA as described in [1] was used. Experiments were repeated incorporating a number of parameter variations. Fifty runs of 300 generations were carried out on each function for each operator.

3 Results

Results showed that UX appears to be the operator which invariably achieves a good (if not the best) quality solution quickly. While BLX achieves the best solution on six problems, it achieves the worst on five others. LX's performance similarly vacillates between being the best and the worst. (Fitness increase is slowest when using BLX.) Significantly, it was on the real-world problems which UX performed the best while LX and BLX performed the worst.

Unsurprsingly, fitness increase corresponded to a proportion of crossovers resulting in children fitter than their parents; however, while some construction occurred, fitness increased regardless of the proportion of destructive crossovers. BLX exhibited a significantly higher proportion of destructive crossovers than constructive (and by far the highest destructive proportion of all the operators) at all times – even on those functions where it was the best-performing operator.

E. Cantú-Paz et al. (Eds.): GECCO 2003, LNCS 2724, pp. 1576–1577, 2003.

3.1 Operators' Effects on Fitness Landscapes

The concept of a fitness landscape was introduced in as a framework for biological evolution; in EC, GAs' recombination operators determine the structure of a fitness landscape. To assess how the operators affected the ruggedness of each test function's corresponding fitness landscape, correlation lengths were measured for each case using the method described in [3].

1PX and 2PX had the longest correlation lengths (low 30s on average); the remaining operators exhibited considerably shorter lengths, with UX and BLX averaging just 5. This suggests that 1PX and 2PX flatten landscapes considerably; the more disruptive operators magnify landscapes' ruggedness by retaining local structure. Clearly, this retention of local structure owes itself to the higher level of exploration effected by the more disruptive operators: many more points on a landscape are reachable via a single application.

4 Conclusion

The empirical evidence presented here appears to question conventional GA wisdom: the results suggest that the assumption that disruption (of schemata) is to be avoided may be flawed (at least in the case of real-coded GAs). Exploration, or more specifically, ergodicity is key to a crossover operator's success. During a run, each variable of a solution is likely to be subjected to fewer changes of value under a less disruptive operator than it would under a highly disruptive one. Moreover, by combining a disruptive operator with a sensible selection strategy the best individuals will automatically be retained while a vigorous exploration of other parts of the search space is carried out [2].

References

1. Drake, S.: Crossover Operators in Real-Coded GAs: An Empirical Study. Cognitive Science Research Paper No. 560. University of Sussex (2003).
2. Eshelman, L. J., Schaffer, J. D.: Real-Coded Genetic Algorithms and Interval Schemata. In: Whitley, D. (ed.): FOGA 2. Morgan Kaufmann (1992).
3. Hordijk, W.: A Measure of Landscapes. Evolutionary Computation Vol. 4 (1996) 335–360.
4. Keane, A.: Experiences with Optimizers in Structural Design. In Parmee, I. C. (ed.): Proc. of the 1st Int. Con. on Adaptive Computing in Engineering Design and Control. University of Plymouth (1994).
5. McIlhagga, M., Husbands, P., Ives, R.: A Comparison of Search Techniques on a Wing-Box Optimisation Problem. In Voigt H.-M., Ebeling, W., Rechenberg, I., Schwefel H.-P. (eds.): PPSN 4. Springer (1996).
6. Syswerda, G.: Uniform Crossover in Genetic Algorithms. In Schaffer, J. D. (ed.): ICGA 3. Morgan Kaufmann (1989).
7. Wright, A. H.: Genetic Algorithms for Real Parameter Optimization. In Rawlins, G.J.E. (ed.): FOGA. Morgan Kaufmann (1991).