problem size n is varied in $[100, 1000]$ (with a step size of 25). In each tested setting, the run is replicated 100 times with different random seeds and the number of function evaluations, denoted as '# evaluations', is reported as the run time. The result for $k = 4$ is shown in Figure 1. Note that we also did the experiment with $p_c = 0$ for the no-mechanism setting, e.g., comparing the EA with the GAs, however, the average run time for $n = 100$ in this experiment is already $2.28 \cdot 10^8$, which cannot be displayed in the figure.

On average, the highest contribution to the reduction of the run time in order is fitness sharing, then convex hull maximisation, deterministic crowding, and, finally, duplicate elimination and minimisation have quite similar average run times. We also notice that the the island model with $\mu = 2$ requires approximately the same average numbers of evaluations as deterministic crowding. Overall, compared to the standard (μ+1) GA, all the diversity mechanisms contribute to the reduction of the average run time, as well as to the stability of the result.

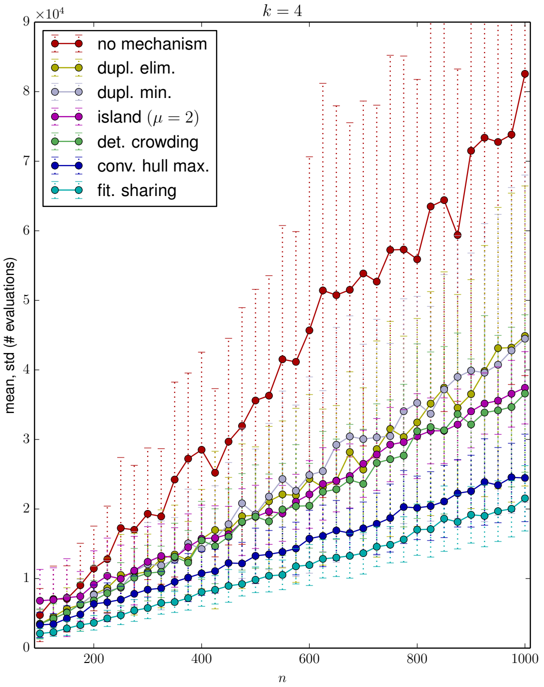

Figure 1: Performance of the diversity mechanisms.

7. CONCLUSION

We have considered the role of selection-based diversity mechanisms used together with crossover for escaping local optima. We prove rigorous upper bounds on the run time of the (μ+1) GA for seven well-known diversity mechanisms optimising the Jump$_k$ function. Our results reveal a qualitative difference in the ability of the different diversity mechanisms to escape local optima.

In contrast to previous theoretical work on crossover for Jump$_k$, our upper bounds do not rely on unreasonably small (e.g., vanishing with n) crossover probabilities, but instead cover the more practical case of constant crossover probabilities. Furthermore, our proofs provide insight into the ways that diversity mechanisms, when applied as a tie-breaking rule in selection, can quickly spread the population out over the jump plateau in order to get enough diversity for crossover to combine the correct solution components to escape the set of local optima.

Acknowledgements

The research leading to these results has received funding from the European Union Seventh Framework Programme (FP7/2007-2013) under grant agreement no. 618091 (SAGE) and from the EPSRC under grant no. EP/M004252/1.

References

[1] E. Alfaro-Cid, J. J. M. Guervós, F. F. de Vega, A. I. Esparcia-Alcázar, and K. Sharman. Bloat control operators and diversity in genetic programming: A comparative study. *Evol. Comput. Journal*, 18:305–332, 2010.

[2] J. Arabas. Approximating the genetic diversity of populations in the quasi-equilibrium state. *IEEE Trans. Evol. Comput.*, 16:632–644, 2012.

[3] G. Bell. *The masterpiece of nature the evolution and genetics of sexuality.* 1982.

[4] E. K. Burke, S. M. Gustafson, and G. Kendall. Diversity in genetic programming: an analysis of measures and correlation with fitness. *IEEE Trans. Evol. Comput.*, 8:47–62, 2004.

[5] N. Chaiyaratana, T. Piroonratana, and N. Sangkawelert. Effects of diversity control in single-objective and multi-objective genetic algorithms. *J. Heuristics*, 13: 1–34, 2007.

[6] B. Doerr and L. A. Goldberg. Adaptive drift analysis. *Algorithmica*, 65:224–250, 2013.

[7] B. Doerr, D. Johannsen, and C. Winzen. Multiplicative drift analysis. *Algorithmica*, 64:673–697, 2012.

[8] T. Friedrich, P. S. Oliveto, D. Sudholt, and C. Witt. Analysis of diversity-preserving mechanisms for global exploration. *Evol. Comput. Journal*, 17:455–476, 2009.

[9] W. Gao and F. Neumann. Runtime analysis for maximizing population diversity in single-objective optimization. In *Proc. of GECCO '14*, pp. 777–784, 2014.

[10] J. He and X. Yao. A study of drift analysis for estimating computation time of evolutionary algorithms. *Natural Computing*, 3:21–35, 2004.

[11] T. Jansen and I. Wegener. The Analysis of Evolutionary Algorithms - a Proof That Crossover really can help. *Algorithmica*, 34:47–66, 2002.

[12] T. Kötzing, D. Sudholt, and M. Theile. How crossover helps in pseudo-boolean optimization. In *Proc. of GECCO '11*, pp. 989–996, 2011.

[13] A. Moraglio and D. Sudholt. Runtime analysis of convex evolutionary search. In *Proc. of GECCO '12*, pp. 649–656, 2012.

[14] F. Neumann, P. S. Oliveto, G. Rudolph, and D. Sudholt. On the effectiveness of crossover for migration in parallel evolutionary algorithms. In *Proc. of GECCO '11*, pp. 1587–1594, 2011.

[15] R. K. Ursem. Diversity-guided evolutionary algorithms. In *Proc. of PPSN VII*, pp. 462–474, 2002.

[16] R. A. Watson and T. Jansen. A building-block royal road where crossover is provably essential. In *Proc. of GECCO '07*, pp. 1452–1459, 2007.

[17] C. Witt. Runtime Analysis of the (μ+1) EA on Simple Pseudo-Boolean Functions. *Evol. Comput. Journal*, 14: 65–86, 2006.