Chapter 8
The Benefits of Population Diversity in Evolutionary Algorithms: A Survey of Rigorous Runtime Analyses

Dirk Sudholt

Abstract Population diversity is crucial in evolutionary algorithms to enable global exploration and to avoid poor performance due to premature convergence. This chapter reviews runtime analyses that have shown benefits of population diversity, either through explicit diversity mechanisms or through naturally emerging diversity. These analyses show that the benefits of diversity are manifold: diversity is important for global exploration and the ability to find several global optima. Diversity enhances crossover and enables crossover to be more effective than mutation. Diversity can be crucial in dynamic optimization, when the problem landscape changes over time. And, finally, it facilitates the search for the whole Pareto front in evolutionary multiobjective optimization.

The analyses presented rigorously quantify the performance of evolutionary algorithms in the light of population diversity, laying the foundation for a rigorous understanding of how search dynamics are affected by the presence or absence of population diversity and the use of diversity mechanisms.

8.1 Introduction

Evolutionary algorithms (EAs) are popular general-purpose metaheuristics inspired by the natural evolution of species. By using operators such as mutation, recombination, and selection, a multiset of solutions — the *population* — is evolved over time. The hope is that this artificial evolution will explore vast regions of the search space and yet use the principle of "survival of the fittest" to generate good solutions for the problem at hand. Countless applications as well as theoretical results have demonstrated that these algorithms are effective on many hard optimization problems.

Dirk Sudholt
Department of Computer Science, University of Sheffield, Sheffield, United Kingdom

© Springer Nature Switzerland AG 2020

B. Doerr, F. Neumann (eds.), *Theory of Evolutionary Computation*,
Natural Computing Series, https://doi.org/10.1007/978-3-030-29414-4_8

A key distinguishing feature from other approaches such as local search or simulated annealing is the use of a *population* of candidate solutions. The use of a population allows evolutionary algorithms to explore different areas of the search space, facilitating global exploration. It also enables the use of recombination, where the hope is to combine good features of two solutions.

A common problem in evolutionary algorithms is *premature convergence*: the population collapses to copies of the same genotype, or more generally, a set of very similar genotypes, before the search space has been explored properly. In this case there is no benefit from having a population; in the worst case, the evolutionary algorithm may behave like a local search algorithm, but with an additional overhead from maintaining many similar solutions.

What we want instead is a *diverse* population that contains dissimilar individuals to promote exploration. The benefits of diversity are manifold:

Global exploration. A diverse population is generally well suited for global exploration, as it can explore different regions of the search space, reducing the risk of the whole population converging to local optima of low fitness.

Facilitating crossover. Often, a diverse population is required for crossover to work effectively. Crossing over two very similar solutions will result in an offspring that is similar to both parents, and this effect can also be achieved by mutation. Many problems where crossover is essential do in fact require a diverse population.

Decision making. A diverse population provides a diverse set of solutions for a decision maker to choose from. This is particularly important in multi-objective optimization as there are often trade-offs between different objectives, and the goal is to provide a varied set of solutions for a decision maker.

Robustness. A diverse population reduces the risk of getting stuck in a local optimum of bad quality. It is also robust with regard to uncertainty, such as noisy fitness evaluations or changes to the fitness function in cases where the problem changes dynamically. A diverse population may be able to track moving optima efficiently or to maintain individuals on different peaks, such that when the global optimum changes from one peak to another, it is easy to rediscover the global optimum.

In the long history of evolutionary computation, many solutions have been proposed to maintain or promote diversity. These range from controlling diversity through balancing exploration and exploitation via careful parameter tuning and designing selection mechanisms carefully, to explicit diversity-preserving mechanisms that can be embedded in an evolutionary algorithm. The latter include techniques such as eliminating duplicates, using subpopulations with migration as in island models, and niching techniques that try to establish niches of similar search points and prevent niches from going extinct. Niching techniques include fitness sharing (where similar individuals are forced to “share” their fitness, i.e., their real fitness is reduced during