This paper uses genetic algorithms augmented with a case-based memory of past problem solving attempts to obtain better performance over time on sets of similar problems. When confronted with a problem we seed a genetic algorithm’s initial population with solutions to similar, previously solved problems and the genetic algorithm then adapts its seeded population toward solving the current problem. We address the issue of selecting “appropriate” cases for injection and introduce a methodology for solving similar problems using genetic algorithms combined with case-based memory. Combinational circuit design serves as a structured testbed and provides insight that is used to validate the feasibility of our approach on other problems. Results indicate that seeding a small percentage of the population with “appropriate” cases improves performance on similar problems and that the combined system usually takes less time to provide a solution to a new problem as it gains experience (memory) from solving other similar problems.

1 INTRODUCTION

Genetic algorithms (GAs) are randomized parallel search algorithms that search from a population of points (Holland, 1975; Goldberg, 1989). We typically randomly initialize the starting population so that a genetic algorithm can proceed from an unbiased sample of the search space. However, we often confront sets of similar problems. It makes little sense to start a problem solving search attempt from scratch with