Abstract

By building an optimization algorithm, one can understand the details of which factors contribute to creating a sufficient optimization algorithm. In this research, a new algorithm was crafted to experiment how a combination of existing optimization techniques may improve performance of the algorithm. The new algorithm, called the Directional Search Algorithm (DSA), utilizes a conglomeration of different techniques normally used in optimization algorithms. These techniques are carefully integrated into one algorithm that geared to finding the global minimum of a single-objective three-dimensional function. DSA is an algorithm derived from an idea that a single search particle extends out in different directions to “scope” out its surroundings. This particle scopes out in multiple directions and finds the solutions at those extended positions. With the incorporation of simulated annealing and a method called “Contraction-Retraction”, the particle will be able to explore the search space expansively. In addition, the algorithm also offers a unique technique that capitalizes an effective balance between exploration and exploitation. The first part of the research will explain and justify the utilization of certain techniques and parameters. The second part of the research will explore the effectiveness of the algorithm. The factors that determine the effectiveness of the algorithm are the following: convergence rate, precision, robustness, and performance. The research examined these four factors on DSA and analyzed the overall performance of the algorithm compared to that of a similar algorithm, the Particle Swarm Optimization (PSO).

Introduction

Optimization problems require immense amount of computational time as the problem scales increase. It is often unrealistic to compute the solution of an optimization problem using brute force because the problem has a nondeterministic polynomial computational time. Thus, metaheuristic algorithms are built to overcome this computational time problem. Although metaheuristic algorithms do not always output the global optimal solution, they provide efficiency in approximating optimal solutions. They make assumptions about the problem space in order to reduce search space and computation.

Background

Simulated Annealing

It is often difficult to create an optimization algorithm because of its variety and complex relationships between parameters. Thus, in many cases, optimization algorithms are often inspired by phenomena in nature. The inspired algorithms are nature proven methods of optimization, and they derive the properties from these natural behaviors that can assist in creating an optimization algorithm that works. One of the most prominent nature-derived method is the “Simulated Annealing”. Its inspiration comes from annealing metallurgy, a technique that involves in cooling the heated material in a controlled manner in order to help the molecules form strong crystals. Each material has its optimum cooling rate to create the strongest form of that material after cooling. Similarly, this method can be used to find a global minimum of a system or a problem. The mathematical function used in simulated annealing determines the probability in which the search will accept worse solution. As the temperature within this function decreases, this probability also decreases. This means that the higher temperature allows for exploration in the initial phase of the search iterations. Once the temperature starts to cool down, the search starts its exploitation and hones down to one area in the search space. Simulated annealing is essentially a tool to balance the exploration and the exploitation within the algorithm. It also offers the probability characteristic that will allow for a more randomized search.

Beam Search

In contrast to simulated annealing method, there is a straight forward concept to pick the solution each step in the optimization algorithm. As simulated annealing offers a probability whether to accept the solution or not, this classic method called “Beam search” is a greedy search algorithm that always accepts the best neighboring node. It explores all possible neighboring nodes and expands to the most promising node. Thus, the probability of accepting a worse solution is always zero. The benefit of the Beam search is that it will approach the solution much quicker than the methods that uses probabilities. Also, another benefit is that it does not have any memory requirement. However, the disadvantage of Beam search is that it may converge quicker into a local optimum will not be able to get out. The Beam search does not explore enough of the search space, and its convergence rate is too high. Thus, once the search enters a local minimum, it has no chance of getting out.

Particle Swarm Optimization (PSO)

PSO is a metaheuristic that utilizes swarm intelligence to find the optimum solution of a system. Swarm intelligence capitalizes the collective effort of multiple particles. The algorithm initiates with a certain number of particles, and it iteratively improves the positions of these particles by using formulas to determine the position and the velocity of each particle.