Grokking Artificial intelligence Algorithms
by Rishal Hurbans

Section 4 Evolutionary Algorithms

When we look at the world around us, we sometimes wonder how everything we see and interact with came to be. One way to explain this is the theory of evolution. The theory of evolution suggests that the living organisms that we see today did not suddenly exist that way, but evolved through millions of years of subtle changes, with each generation adapting to its environment. This implies that the physical and cognitive characteristics of each living organism are a result of best fitting to its environment for survival. Evolution suggests that organisms evolve through reproduction by producing children of mixed genes from their parents.

Evolution

Evolution produces variation in a population and the dynamic environment choses what variants are better suited for living in it. This concept can be applied to evolutionary algorithms

Knapsack problem

_____Evolutionary algorithms are not typically used to solve problems, rather they are mainly used for optimization problems where there are many solutions with some being more optimal. As seen in the knapsack problem where a given “knapsack has a specific maximum weight that it can hold. Several items are available to be stored in the knapsack, and each item has a different weight and value. The goal is to fit as many items into the knapsack as possible so that the total value is maximized and the total weight does not exceed the knapsack’s limit. The physical size and dimensions of the items are ignored in the simplest variation of the problem” Although this problem seems simple with low cost light items, the more items with higher value and weight cause the tree computational cost to become massive. Too massive to try a brute-force method. Therefore one can use the Genetic Algorithm to try to find an optimal solution. “It is important to note that a genetic algorithm is not guaranteed to find the absolute best solution; it attempts to find the global best while avoiding local best solutions.” The genetic algorithm life cycle can be broken down to the following: Creating a random population of potential solutions, determining how good a specific solution is, selecting parents based on fitness to produce offspring, reproducing individuals by mixing genetic information and applying mutations, and selecting individuals from the population that will survive to the next generation. You can generate an individual with random genes and determine if this solution is valid based on its fitness score. We can use fitness functions to determine how good they are, they are similar to heuristics seen in the past chapter. The fitness function measures the total value of items in the knapsack for each

individual. Those with higher total values would be more fit. To choose the next set of parents a common approach is to use the roulette-wheel selection method. Involving where individuals are given different portions of a wheel based on fitness, the wheel is spun and an individual is selected. Higher fitness individuals with a higher portion of the wheel are more likely to be selected, these parents can created one or two mixed children. With two point crossover, two points on the arrays of the parents are chosen, creating 3 sections in each parent. The two middle portions of each parent are switched, creating 2 new mixed children. Uniform crossover has similar ideas to two point crossover and is believed to create more diverse children. With bit string mutation, a random index is selected randomly and changed to a different valid value. With flip bit mutation all values are inverted.” The beauty of this situation is that the algorithm explores as much of the search space as possible while exploiting strong solutions as individuals evolve.”

Important figures

Knapsack Problem

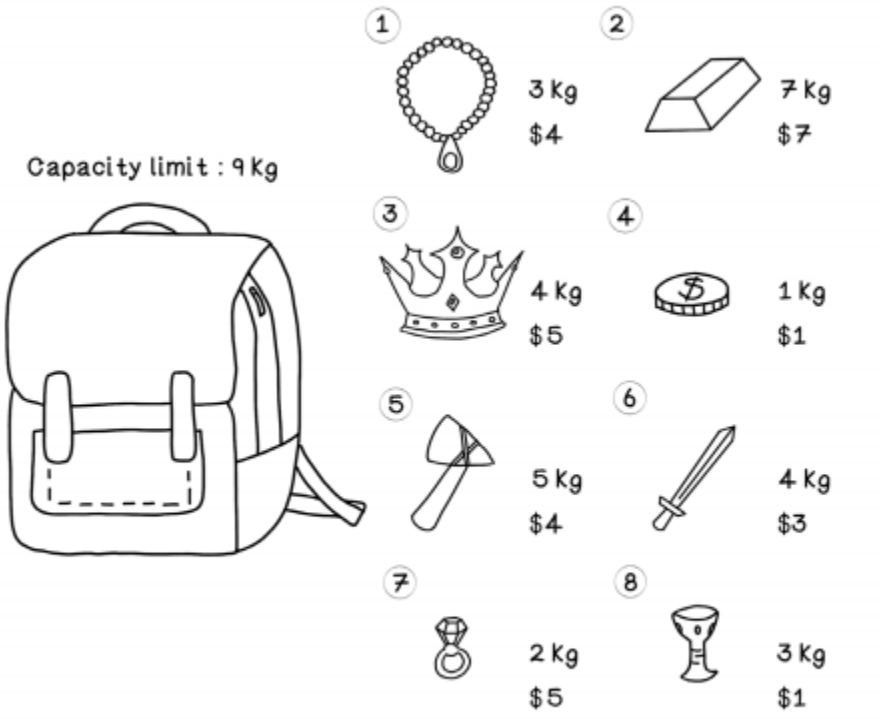

Figure 4.4 A simple Knapsack Problem example

Local and Global best

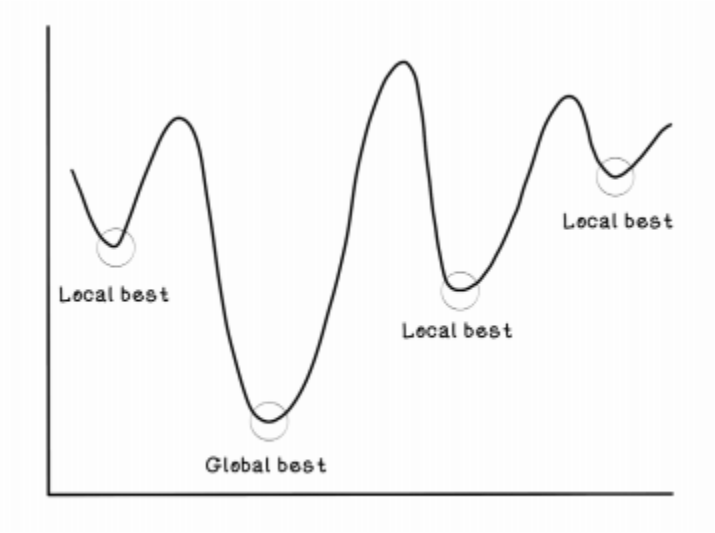

Figure 4.7 Local best vs. global best

Convergence

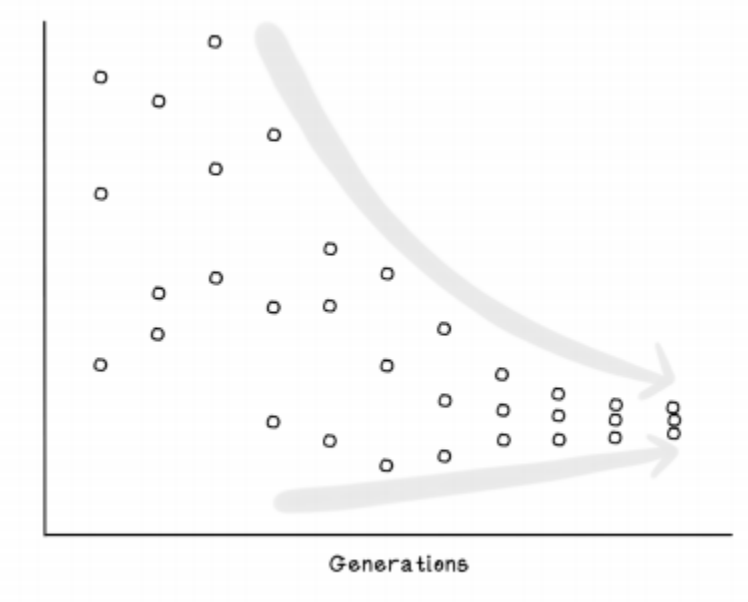

Figure 4.8 Diversity to convergence