Syracuse University
College of Engineering
& Computer Science

# CIS 600: Evolutionary Machine Learning

## Project Proposal

Title: *Application of MCTS methods on Genetic Algorithms to guide the selection heuristic*

*Akshay Hebbar*

# Introduction

When solving complex optimization problems, genetic algorithms (GA) can be an effective heuristic search technique for finding high-quality solutions. However, the performance of GA is often heavily dependent on the quality of the population and the selection of individuals for genetic operators such as mutation and crossover. To address these issues, we are suggesting a novel approach of combining GA with other optimization techniques such as Monte Carlo Tree Search (MCTS). One way to integrate MCTS with GA is to use MCTS to guide the selection of individuals for the genetic operators. By using MCTS to identify promising regions of the search space, the selection phase of the GA can be optimized to focus on these areas and improve the quality of the solutions generated by the algorithm. This approach has the potential to enhance the performance of GA and provide a more data-driven approach to solution generation and evaluation.

# Proposed Architecture

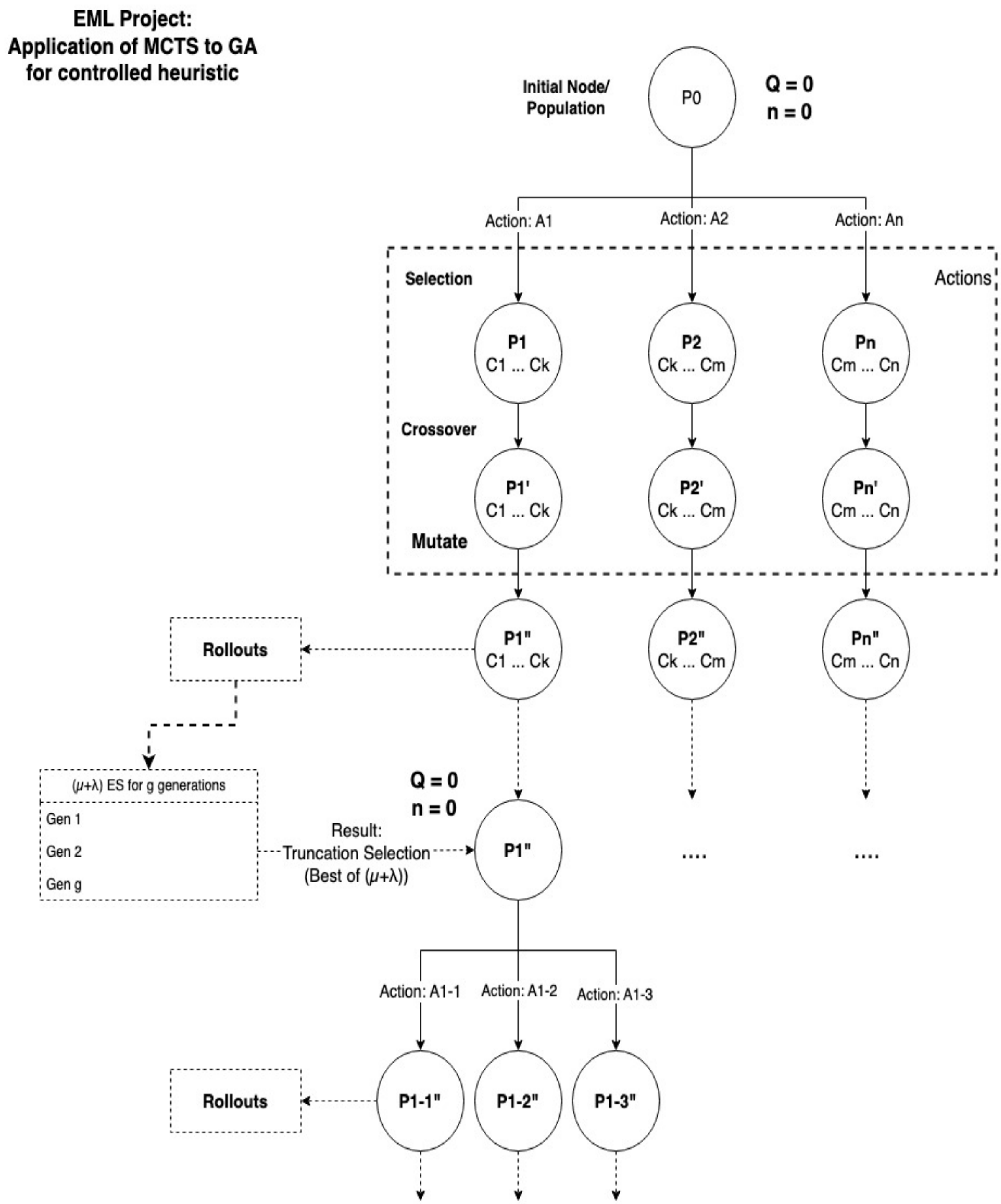


Figure 1: MCTS process as applied to (GA + ES)

# Algorithm

We start with a general initial population P0 of p individuals, as depicted in figure 1. The population is initialized to the root node, and the corresponding Q value and node visit count are set to 0.

To generate the children of the root node, we apply actions Ai on the initial population P0. Each action Ai consists of a sequence of steps - selection, crossover, and mutation - as a way of applying genetic algorithm on the parent population and produces a series of mutated children population. The mutation of the parent population helps to increase the diversity of the population as well as explore possible solutions.

Then, we apply UCB1 or a similar approach to choose the child node in such a way as to allow for balanced exploration and expansion of the tree.

The tree is traversed until a leaf node is obtained. At the leaf node, we apply a rollout process of simulation, where we use $(\mu + \lambda)$ evolutionary strategy for the mutated child with recombination and self-adapting $\sigma$.

Here, we can choose to control the number of generations the evolution occurs to improve the accuracy of the fitness. Once we reach a specified threshold, we replace the mutated child population with the result of the rollout.

The Q value for the node is then calculated as the average fitness of the node population obtained from the rollout, and the visit count is incremented.

Finally, we repeat the process until a desired tree height is reached. This method allows us to apply Genetic Algorithm and Evolutionary strategy in unison while providing a backbone to direct our selection heuristic towards likely solutions using the MCTS as the backbone.