Assignment2: Robot Task Optimization Using Genetic Algorithm

This report presents a Python implementation of a Genetic Algorithm (GA) for task assignment optimization. The GA aims to allocate tasks to robots based on task durations, priorities, and robot efficiencies. The code is divided into functions for generating mock data, defining fitness functions, selecting parents, performing crossover and mutation operations, running the GA, and visualizing task assignments.

The generate_mock_data function generates random arrays for tasks and robots, while the fitness function calculates the fitness of each individual in the population. The select_parents function randomly selects potential parents based on their fitness, while the crossover function performs single-point crossover to create children from selected parents. The mutation operation introduces random changes to the children's genes, ensuring exploration of new solutions in the search space.

The run_genetic_algorithm function orchestrates the GA process, initializing a random population, iterating over generations, evaluating fitness, selecting parents, performing crossover and mutation, and updating the population. The visualize_assignments_improved function visualizes task assignments on a grid, displaying task durations with annotations indicating task priorities and durations for each assigned task on respective robots.

In conclusion, this Python implementation provides a framework for solving task assignment optimization problems using Genetic Algorithms, efficiently allocating tasks to robots based on task characteristics and robot efficiencies. Further enhancements could involve fine-tuning parameters, implementing alternative crossover and mutation strategies, and evaluating performance on diverse datasets. Overall, this code serves as a valuable tool for optimizing task allocation in various domains.