Computational Intelligence

Assignment One

* Initialization: Begin by generating a population of random solutions within the search space of fixed size. Ensure that each solution is valid according to the problem constraints.
* Fitness Evaluation: Evaluate the fitness of each individual in the population. In this scenario, we count the number of ones in each individual's chromosome, assuming a binary representation. This fitness function aims to maximize the number of ones and is tailored to the specific problem at hand.
* Selection: Employ roulette wheel selection to choose potential parents for the next generation. Calculate each individual's probability by summing their fitness values and dividing each fitness by the total. Then, compute the cumulative probability distribution. Higher fitness individuals have a greater chance of being selected as parents.
* Crossover: Apply one-point crossover to create new offspring. Determine a crossover rate, a probability that governs whether crossover occurs for a given pair of parents. Additionally, select a random point along the chromosomes of the parents to exchange genetic material and generate new offspring.
* Mutation: Perform bit-flip mutation on the offspring with a predefined mutation rate. This involves randomly flipping bits in the chromosome, introducing genetic diversity into the population. Adjust the mutation rate according to the problem requirements.
* Population Replacement: Form the new population by combining the offspring generated from crossover and mutation with some individuals from the previous generation. Use elitism to retain the best solutions from the previous generation in the new population, ensuring that the overall quality of solutions does not degrade over generations.
* Termination: Repeat these steps until a termination condition is met, such as reaching a maximum number of generations or finding a satisfactory solution that meets the optimization criteria.

By following these steps, the genetic algorithm iteratively searches for optimal or near-optimal solutions within the given search space.