AI311 – Nature Inspired Computing and Optimization

Practical 2: Lab Sheet

Title:

Implementation and Analysis of Greedy and Randomized Heuristic Search for Discrete Optimization

Aim:

To implement, evaluate, and compare heuristic search techniques — Greedy Hill Climbing and Randomized Search — for identifying the minimum of a discrete function.

Problem Statement:

Given the discrete objective function:

$f(x) = x^2 - 6x + 8$, where $x \in [-10, 10]$ and x is an integer,

use heuristic approaches (excluding brute force) to locate the minimum.

Tasks:

1. Greedy Hill Climbing Search

- Choose two starting integers: one near the center and one near a boundary.
- In each iteration:
  - Evaluate neighbours: $f(x - 1)$ and $f(x + 1)$.
  - Move to the neighbour with a lower function value.
- Stop when no further improvement is possible (i.e., a local minimum is reached).
- Record the sequence of x values visited.

2. Randomized Search (Stochastic Local Search)

- Start from a randomly selected integer $x_0 \in [-10, 10]$.
- At each step:
  - Randomly pick either $x - 1$ or $x + 1$, staying within bounds.
  - Move to the chosen neighbour only if it improves the function value.
- Run for a fixed number of steps (e.g., 25).
- Repeat the experiment at least three times with different random starting points.
- Record the search paths for each trial.

3. Visualization

- Plot the function $f(x)$ over the domain $x \in [-10, 10]$.
- Overlay the search trajectories from both methods.
- Highlight the minima discovered by each method.

4. Discussion

- Compare both approaches in terms of:
  - Quality of the solution (final value)
  - Convergence behaviour (speed and stability)
- Discuss scenarios where greedy search may miss the global minimum.
- Explain how randomness can help escape local minima or flat regions.
- Analyse the role of initial positions in influencing search outcomes.

5. Sensitivity to Initialization – Multiple Start Points in Greedy Search

Objective:
Evaluate how the starting point affects the outcome of Greedy Hill Climbing.

Instructions:
- Perform the Greedy Hill Climbing algorithm from at least five different starting points: x = -10, -5, 0, 5, 10
- Record the final x and f(x) values.
- Compare all results:
  - Which start point leads to the global minimum?
  - Which ones are stuck in local minima?

Expected Output:
- Table showing all start points → final result
- Visualization: overlay all search paths on the function curve

6. Hybrid Search (Greedy + Random Jump)

Objective:
Implement a basic random-restart hill climbing algorithm to escape local minima.

Instructions:
- Start with Greedy Hill Climbing.
- If stuck (no better neighbor), restart from a new random x_0.
- Allow max 5 restarts or until the global minimum is found.
- Visualize and report how many restarts were needed.

7. Factory Optimization Problem (Bonus Task)

Objective:
Solve a constrained optimization problem using mathematical formulation and optionally a solver.

Scenario:
- A factory produces two products: A and B.
- The profit per unit of A is $40, and for B it is $30.

- Producing one unit of A requires 2 hours of machine time and 3 kg of raw material.
- Producing one unit of B requires 1 hour of machine time and 4 kg of raw material.
- The factory has 100 hours of machine time and 240 kg of raw material available per week.

Task:
- Formulate the problem as a linear programming model to maximize profit.
- Solve it manually or using any LP solver (optional).
- Discuss the optimal quantities of A and B and total profit.