CMPS 403/CMPE 470 Artificial Intelligence

Project Part II

Project: Solving the N-Queens Problem Using Constraint Satisfaction Techniques

Objective

Design and implement a system to solve the N-Queens Problem using Constraint Satisfaction Problem (CSP) techniques. The goal is to understand and compare different approaches for solving CSPs in terms of efficiency and effectiveness. You will visualize solutions using a graphical board.

Problem Description

The N-Queens Problem asks:
How can N queens be placed on an N×N chessboard so that no two queens attack each other?

A valid solution must ensure:
- No two queens share the same row, column, or diagonal.

Project Requirements

You must design, implement, and compare the following four techniques for solving the N-Queens problem:

1. Naïve Approach

Represent the board as a list or matrix.
Try all possible placements of queens.
Check for conflicts manually (no two queens attack each other).
Report the number of valid solutions found.
Analyze the time complexity.

2. Backtracking with Forward Checking

Implement a standard backtracking search.
Use forward checking to prune inconsistent values from the domains of unassigned variables after each assignment.
Track the number of consistency checks or node expansions.
Implement additional heuristics such as MRV (Minimum Remaining Values)

3. Backtracking with Arc Consistency (AC-3)

Extend your backtracking algorithm to maintain arc consistency (e.g., using the AC-3 algorithm) after each assignment.

Compare how this approach performs compared to forward checking.
Implement additional heuristics such as MRV (Minimum Remaining Values)

4. Randomized Local Search (e.g., Min-Conflicts)

Implement a Min-Conflicts heuristic:
- Start with a random assignment.
- Iteratively move a queen to reduce the number of conflicts.
Discuss how performance changes with larger N.

Deliverables

1. Code Implementation for all four methods.
When the code is run it should output the solution for all the four methods, number of explored nodes, checked final assignments, and running times.
2. Experimental Results:
  - Compare runtime performance and number of steps for each method on different N values (e.g., N = 4, 8, 16, 32).
3. Report (3–5 pages) including:
  - Problem formulation as a CSP (Variables, domains, and constraints).
  - Description of each method.
  - Comparison of performance (include tables or graphs). Comparison should include the number of explored nodes and the running time for various N values.
  - Discussion of advantages and limitations of each approach.

Submission

You should submit a compressed directory that has two subdirectories, named 'project1' and 'project2'. The directory's name should be GroupID_Project.

Project Discussion

Discussion of the report and the code is an integral part of the project's evaluation. The discussion will count towards **one third** of the project's final grade (5 points towards the course's final grade).