Report: Solving the 8-Queens Problem using Hill Climbing Algorithm Variants

1. First Choice Hill Climbing:

Description: First Choice Hill Climbing is a variant of the Hill Climbing algorithm that aims to find the best solution by making incremental improvements. In this variant, instead of always choosing the best neighbor, it selects a random neighbor and moves to it if it leads to a better solution. This randomness helps in escaping local optima.

Application to 8-Queens Problem: For the 8-Queens problem, First Choice Hill Climbing works by placing queens on the chessboard randomly and then making small modifications to the arrangement to minimize conflicts. It chooses a queen randomly and moves it to another row within its column if it reduces the number of conflicts.

```
Pseudo Code:
function first_choice_hill_climbing(problem):
    current_state = problem.initial_state
    while true:
        neighbors = problem.generate_neighbors(current_state)
        if no neighbors have fewer conflicts than current_state:
            return current_state
        next_state = randomly select a neighbor
        if next_state has fewer conflicts than current_state:
            current_state = next_state
```

Performance Analysis:

- Strengths:
  - Can escape local maxima better due to randomness.
- Weaknesses:
  - May not always find optimal solutions.
  - Can be slow if the randomly chosen neighbors are not better.

Time Complexity: The time complexity depends on the number of iterations needed to find a solution. In the worst case, it could be exponential if the algorithm gets stuck in a local maximum.

Probability of Success: The probability of success varies based on the initial state and the randomness of the algorithm. In general, it has a higher chance of success than basic Hill Climbing due to the random selection of neighbors.

2. Steepest Ascent Hill Climbing:

Description: Steepest Ascent Hill Climbing is a variant where at each step, it evaluates all neighboring states and moves to the one that maximizes the improvement in the evaluation function. It differs from First Choice Hill Climbing by considering all neighbors and choosing the best one.

Application to 8-Queens Problem: In the 8-Queens problem, Steepest Ascent Hill Climbing evaluates all possible moves for each queen and selects the one that results in the fewest conflicts. It systematically explores the search space to find the best solution.

Pseudo Code:

```
function steepest_ascent_hill_climbing(problem):
    current_state = problem.initial_state
    while true:
        neighbors = problem.generate_neighbors(current_state)
        next_state = the neighbor with the fewest conflicts
        if next_state has fewer conflicts than current_state:
            current_state = next_state
        else:
            return current_state
```

Performance Analysis:

- Strengths:
  - Tends to find better solutions compared to First Choice Hill Climbing.
- Weaknesses:
  - Can still get stuck in local maxima.
  - Can be computationally expensive due to evaluating all neighbors.

Time Complexity: Similar to First Choice Hill Climbing, but potentially higher due to evaluating all neighbors at each step.

Probability of Success: Higher probability of success compared to First Choice Hill Climbing due to considering all neighbors.

3. Random Restart Hill Climbing:

Description: Random Restart Hill Climbing is an approach where the algorithm restarts multiple times from different initial states. This helps in exploring different parts of the search space and avoiding getting stuck in local maxima.

Application to 8-Queens Problem: In the 8-Queens problem, Random Restart Hill Climbing involves running Hill Climbing multiple times with different initial configurations of queens. It restarts the algorithm until it finds an optimal or satisfactory solution.

Pseudo Code:

```
function random_restart_hill_climbing(problem, num_restarts):
    best_solution = null
    for i from 1 to num_restarts:
        current_solution = hill_climbing(problem)
        if current_solution is better than best_solution:
            best_solution = current_solution
    return best_solution
```

Performance Analysis:

- Strengths:
  - More likely to find global optimum by exploring different areas of the search space.
- Weaknesses:
  - Can be computationally expensive due to multiple restarts.
  - Might still get stuck in local optima in each run.

Time Complexity: Depends on the number of restarts and the complexity of the Hill Climbing algorithm. Overall, it's higher than basic Hill Climbing due to multiple runs.

Probability of Success: Higher than basic Hill Climbing due to exploring different initial states.

4. Stochastic Hill Climbing:

Description: Stochastic Hill Climbing is a variant where instead of selecting the best move deterministically, it randomly selects a move according to some probability distribution.

Application to 8-Queens Problem: In the 8-Queens problem, Stochastic Hill Climbing involves randomly selecting a move from the set of all possible moves with probabilities based on their quality. It introduces randomness to the search process.

```
Pseudo Code:
function stochastic_hill_climbing(problem):
    current_state = problem.initial_state
    while true:
        neighbors = problem.generate_neighbors(current_state)
        next_state = randomly select a neighbor based on some
probability distribution
        if next_state has fewer conflicts than current_state:
            current_state = next_state
```

Performance Analysis:

- Strengths:
  - Can escape local maxima better due to randomness.
- Weaknesses:
  - May not always find optimal solutions.
  - Requires careful tuning of probability distributions.

Time Complexity: Similar to First Choice Hill Climbing, but potentially higher due to the randomness in selecting neighbors.

Probability of Success: Similar to First Choice Hill Climbing but can be influenced by the choice of probability distributions.

5. Experimental Analysis:

Experimental analysis involves conducting experiments to evaluate the performance of each algorithm variant in solving the 8-Queens problem. Metrics such as solution quality, convergence speed, and computational efficiency should be measured and presented using tables, charts, or graphs for clarity.

6. Discussion and Conclusion:

In conclusion, each Hill Climbing variant has its strengths and weaknesses in tackling the 8-Queens problem. First Choice Hill Climbing offers randomness to escape local maxima but may not always find optimal solutions. Steepest Ascent Hill Climbing considers all neighbors for each move, providing better solutions but can be computationally expensive. Random Restart Hill Climbing explores different areas of the search space but requires multiple runs. Stochastic Hill Climbing offers randomness in selecting moves but requires careful tuning of probabilities. Overall, the choice of algorithm depends on the specific requirements of the problem and the trade-offs between solution quality, computational resources, and time

constraints. Further improvements could involve hybrid approaches combining different variants or employing more sophisticated search techniques.

In [ ]: