****

**INDEPENDENT UNIVERSITY,**

**BANGLADESH**

**Artificial Intelligence**

**Report**

**Team Project**

**Topic: Hill Climbing and Swarm Intelligence Algorithms**

**Context: Solving the Travelling Salesman Problem**

**Faculty: Amin Ahsan Ali, Associate Professor, Dept of CSE, IUB**

**Email:** [aminali@iub.edu.bd](mailto:aminali@iub.edu.bd)

**Prepared by  
Fahim Shahriar Eram (2022523), Syed Niaz Mohtasim (2021607)**

**Date of Submission: 16th August 2023**

**Semester: Summer 2023**

# The Travelling Salesman Problem and its applications

Travelling Salesman Problem (TSP) is a classic optimization problem in computer science and mathematics. It asks for the shortest possible route that visits each city exactly once and returns to the starting city.

TSP has many applications in different fields, such as:

* Planning and logistics, such as vehicle routing, aircraft route arrangement, and job sequencing.
* Manufacturing, such as computer wiring and cutting wallpaper.
* Biology, such as DNA sequencing.
* Artificial intelligence, such as using intelligent algorithms to solve TSP.

# Objectives of this project

The main goal of this project is to solve TSP by using two different algorithms, the Hill climbing algorithm and a swarm intelligence algorithm of our choice. We chose to go with the Bat Algorithm. The Bat Algorithm strikes a balance between exploration (searching the solution space for diverse solutions) and exploitation (refining promising solutions).

# Description of the Algorithms used

**Hill climbing algorithm** is an optimization algorithm that tries to find the best solution for a given problem by moving in the direction of increasing value until it reaches a peak or a local maximum. It is a type of local search algorithm that is often used in artificial intelligence and to solve TSP.

On the other hand, the **Bat algorithm** is a metaheuristic optimization algorithm that mimics the echolocation behavior of Bats when they search for their prey. It is based on the idea that Bats can adjust their pulse emission rate and loudness according to their distance to the target. The Bat algorithm can be applied to solve the TSP by using a discrete representation of the solution, where each Bat represents a possible tour of the cities.

# Experimental Setup

We used Python as the programming language to implement these algorithms and the Visual Studio Code IDE as our code editor. We used libraries such pandas and numpy for data reading and processing. We also used matplotlib library for the graph.  
We modelled TSP as an undirected weighted graph, such that cities are the graph's vertices, paths are the graph's edges, and a path's distance is the edge's weight. It is a minimization problem starting and finishing at a specified vertex after having visited each other vertex exactly once. Often, the model is a complete graph.

For each algorithm, we ran it a certain 100 number of times (100 runs) with the same set of problem instances. We ran both algorithms on 3 different instances of symmetric TSP (distance from city A to city B is same as city B to city A) which we obtained from the TSPLIB library. The 3 file instances are berlin52, att48 and pr76.

The maximum iteration was also set at 100 for Hill-climbing algorithm and set to 5000/population size for Bat algorithm to fulfill the requirement of the program not running for more than 5 minutes. The population size was varied from 50 to 100 to 250 and finally 500.

The performance metric we used to evaluate the algorithms is the total distance travelled by the tour, we chose the shortest distance from the 100 runs and chose it as our best distance and compared it to the optimal distance, which we got from TSPLIB as well. We used these two distances to calculate an error percentage for both algorithms.

# Results

For Hill climbing (runs = 100, max iterations = 100),

Error Percentage for berlin52 = 123.54%

Error Percentage for att48 = 141.6%

Error Percentage for pr76 = 237.39%

Graphs for Bat Algorithm, Error vs Population Size (max iterations = 5000/population\_size)

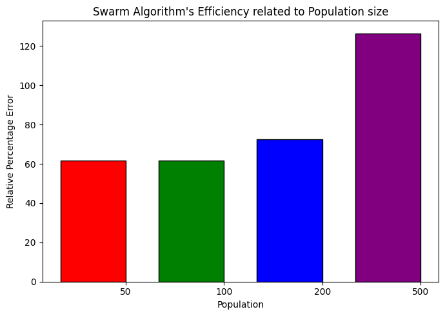
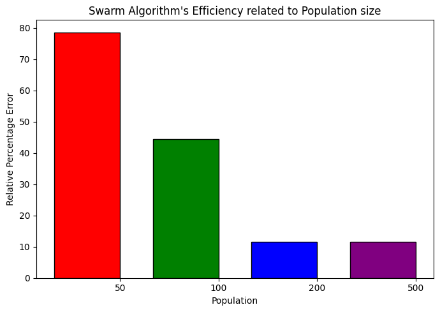
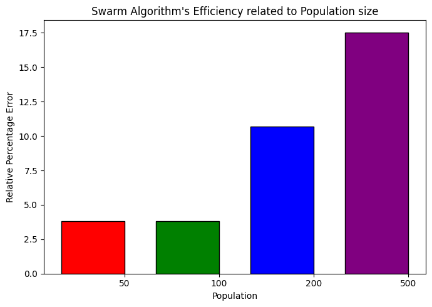
  

Figure berlin52 Figure 2 att48 Figure 3 pr76

# Conclusion

In conclusion, our comparative study of the Hill Climbing and Bat algorithm in solving the Traveling Salesman Problem (TSP) provides valuable insights into their respective strengths and limitations.

While the Hill Climbing approach showed limitations in escaping local optima and achieving consistent convergence to optimal solutions, it did highlight the simplicity and ease of implementation for relatively straightforward optimization tasks.

The Bat algorithm's adaptability to different population sizes showcased its versatility and potential for optimization across various problem complexities. However, there is an element of randomness in Bat algorithm due to the selection of values of position and velocities, frequency, pulse emission being randomised at the beginning. However, due to this randomness it also ends up helping the algorithm escape local optima, explore diverse solutions, and avoid getting stuck in a single trajectory. The balance between randomness and exploitation is crucial to achieving effective optimization performance.

In terms of future research, exploring hybrid algorithms that combine the strengths of both Hill Climbing and the Bat Algorithm could be promising. Furthermore, investigating the Bat Algorithm's performance on more complex optimization problems and its scalability to larger problem instances remains an avenue for future exploration. We would have liked to run it for more iterations but refrained from it due to the requirement and running it does take up significant time and computational resources.

Ultimately, our findings suggest that the Bat algorithm is a promising choice for solving the TSP due to its adaptive nature, ability to generate diverse solutions, and consistent outperformance in terms of solution quality.

# Contributions

In the completion of this project, our team, consisting of Syed Niaz Mohtasim and Fahim Shahriar Eram collaborated closely to successfully implement and evaluate two prominent optimization algorithms, namely Hill Climbing and the Bat Algorithm, for solving the Traveling Salesman Problem (TSP). The project was divided into distinct roles, where each member contributed their expertise to different facets of the project but also were aware at all times of what each member were doing and both have a good understanding of what the other did.

Fahim worked on the algorithm implementations and coding, wrote the section of report on TSP and its applications.

Syed worked on graph generation and the rest of the report.