BIU Journal of Basic and Applied Sciences 10(1): 87 – 98, 2025.

ISSN: 2563-6424

OPTIMIZING TRAVEL ROUTES IN EDO STATE WITH GENETIC ALGORITHM MODEL

***OTAREN, F. E., IMIANVAN, A. A. AND AMADIN, F. I.**
Department of computer Science, University of Benin, Benin City, Nigeria
*Corresponding author: fredrick.otaren@physci.uniben.edu

ABSTRACT

Technological revolution has contributed immensely to the increase of computer information, growing computational capabilities of devices, raising the level of knowledge abilities and skills and increasing developments in science and technology. This study focused on designing a genetic algorithm model for finding the shortest path of a travelling salesman within cities in Edo State. For this purpose, existing algorithms like cuckoo were reviewed, and genetic algorithm was used to develop the model. The algorithm was tested on predefined set of cities in Edo State and implemented using C# programming language. The dataset of the predefined cities was collected using Google Map. Another approach involved the use of GPS, Open Street Map, and historical Data on travel times or distance between two cities which can be collected from various sources such as Government reports, Transportation Authorities or commercial database. The dataset collected was used to construct and populate the distance matrix. After the review of existing algorithm, genetic algorithm was used to design a model that solved the Travelling Salesman Problem (TSP), and the implementation was done withC#. NET Winforms. The result of the study was based and evaluated on two criteria: the minimum distance and computing time.

KEYWORDS: *Traveling Salesman Problem, Optimization Algorithms, Edo State Logistics, Distance matrix, Google map*

INTRODUCTION

Technological revolution has contributed immensely to the increase of computer information, growing computational capabilities of devices, raising the level of knowledge abilities and skills and increasing developments in science and technology. Several algorithms have been developed in the past decades for solving the travelling salesman and the shortest path problem. The shortest path can be single pair shortest path problem or all pairs shortest path problem.

The travelling salesman problem (TSP) is a well-known computational problem in which the objective is to find the shortest possible route that visits a set of cities or locations only once. It has numerous practical real-world applications that include designing delivery route for courier services, tour planning for tourists, circuit board drilling, optimizing production schedules in manufacturing, logistics, transportation and planning school bus route.

Edo State like many other cities around the world, with a large landmass and a growing economy faces the challenge of optimizing travel route for delivery companies, transportation systems and other logistics related businesses. There are many cities that are important for business and commerce, finding the shortest path among them can help to optimize transport and logistics operations. However, as the state population continues to grow and urbanize with large number of cities and a diverse geography, finding the shortest path using manual method can be time consuming and prone to errors, hence optimizing travel routes is a critical challenge that can benefit from the application of TSP algorithms. The TSP has many real-life applications in various areas such as planning, logistics, machine scheduling problems, the manufacture of microchips, and DNA sequencing, Petrica *et al*. (2024).

There have been significant advances in optimization algorithms that can be used to solve the TSP. These include genetic algorithms, ant colony optimization, simulated annealing and tabu search among others. These algorithms have been successfully applied to a wide range of optimization problems including the TSP. A good and recent algorithm for solving the travelling salesman problem is the Cuckoo Search Algorithm with levy flights (CSA-LF). It is a recently proposed meta-heuristic algorithm that is based on the behavior of cuckoo birds. It was first introduced by Yang and Deb in 2010, and has been shown to be effective at solving TSP. the algorithm iteratively improves a set of candidate solutions by generating new solutions using levy flights, and replaces the worst solutions with new ones. TSP is a challenging optimization that has been studied extensively in literature. While many algorithms have been proposed for solving TSP, each algorithm has its own strength and weakness. The choice of algorithm depends on the specific problem instance and requirements. The Greedy Algorithm (GA) is also a simple heuristic algorithm that is based on the idea of selecting the locally optimal solution at each step. GA has been used for solving the shortest path problem in various contexts, including transportation networks, communication networks, and logistics. For example, Zhao (2024) described the realization principle of intelligent search algorithm such as genetic algorithm in dynamic routing optimization of computer network.

Wang *et al*. (2021) proposed a multi-path routing algorithm for WSN based on genetic algorithm. The fitness function was determined by calculating the node spacing, and a shared routing scheme was generated at the base station. Because of the fixed-length coding, the best path may be limited, and the global optimization cannot be achieved. In the transportation demand modeling field, novel approaches appear, which aim to optimize the scheduling of individuals or goods. These approaches are developed as combinatorial optimization problems based on the Traveling Salesman Problem (TSP), Esztergár-Kiss *et al*. (2020).

Many different approaches have been proposed to address MTSP including Ant Colony Optimization (ACO) and GA which belong to nature-inspired approaches. ACO approaches are used to produce solutions that use the 2-opt edge exchange approach to improve the solutions obtained. The basic concept of using ACO to solve the MTSP is the salesmen on their tours simulate or act like ants in their search for food, they go separately leaving a pheromone in the path they follow while searching for food, Harrat *et al.* (2019). Mukhairez and Maghari, (2015) compared performance and efficiency of 3 optimization techniques in solving the TSP. Their TSP involved 30 cities giving a total possible number of combinations of ~ 2.7 x 10^{32} which is an 8enormous number. The techniques they compared were Simulated Annealing (SA), a technique that models the annealing process of materials from the field of metallurgy, Genetic Algorithm and Ant Colony Optimization. They observed both the execution times of each algorithm as well as the optimal result achieved by the respective algorithms. They reported that in terms of best result (shortest path) ACO was the clear winner but also had the slowest execution time. SA had the fastest execution time and had comparable results to GA, while GA came in second place for both execution time and result. Batista *et al*. (2015), in a research used the greedy randomized adaptive search procedure to find the shortest path for a traveling salesman between 100 cities in Brazil. The results show that the greedy algorithm performed better than the simulated annealing algorithm and the genetic algorithm in terms of finding the shortest path.

Deng *et al*. (2021) proposed the use of a modified GA based on cellular automata and simulated annealing algorithm to solve the TSP. First, the selection operation of the modified GA is applied, and then the GA is combined with the simulated annealing algorithm to search for the best tour. The proposed approach is tested using 13 instances from the TSBLIB. Good results are achieved compared to the other algorithms. Alazzam *et al*. (2020) solved the MTSP by a meta-heuristic technique called discrete Pigeon-Inspired Optimizer (PIO). The total cost of the shortest tour for active salesmen with load balancing is computed. The merit of this approach is the ability to find the solution based on pigeon operators, landmark, map, and compass operators. Map and compass operators depend on both the position and velocity of salesmen, but landmark used salesmen's landmark neighboring. The PIO was tested using five instances of the TSPLIB. The experimental results showed that the PIO outperformed all compared algorithms within the most used instances. The best results are obtained with 2 and 3 salesmen for all instances

Rani *and kumar,* (2014), proposed the improved genetic algorithm by a roulette-wheel selection operator with different crossovers and mutation rates to solve the traveling-salesman problem. Experimental results demonstrated that the algorithm in this paper is better than the existing crossover operator in other algorithms, Rufai *et al*. (2021) asserted that an Improved Ant Colony Optimization Algorithm was able to find near-optimal solutions for TSP up to 56 cities, when it was applied to TSPs in Nigeria. The application of TSP has been explored to specific domains and contexts, such as urban planning, environmental management, and tourism

Yu *et al*.2016, presented a hybrid simulated annealing algorithm based on tabu search to solve the traveling-salesman problem. Experimental results demonstrated that the proposed algorithms improved accuracy and efficiency also Absalom *et al.* (2017) presented a simulated annealing algorithm based on symbiotic-organism search in order to better solve the traveling-salesman problem. Comparative results showed that the proposed algorithm had advantages in terms of convergence, average execution time, and percentage deviations. El-Samak *et al*. (2015) applied the affinity propagation clustering technique to optimize the genetic algorithm for finding the best solutions to the traveling-salesman problem. Mohsen *et al*. (2016) proposed a hybrid ant-colony optimization algorithm based on the simulated annealing algorithm to solve the traveling-salesman problem. He *et al*. (2014) put forward an improved ant-colony algorithm based on new crossover and mutation operations to find the best solution for the traveling-salesman problem. Simulation results demonstrated that the proposed algorithm had advantages in terms of stability and optimization capacity.

METHODOLOGY

This research methodology provides a systematic approach to developing and assessing the proposed Genetic Algorithm Model, ensuring a robust and thorough investigation of its effectiveness in the specific context of Edo State. In the proposed model, a genetic Algorithm was used to design the model that can optimize the shortest and best route of a traveling salesman, and the implementation was done using C# programming language.

The overall activities constituting the study are represented as follows;

1. Review of Existing Algorithm such as the Cuckoo Search Algorithm, commonly used in solving optimization problem.

ii. Define the problem, specifying the list of cities or locations within Edo State, including Oredo, and the distances or travel times between them.

iii. Gather data on the distances or travel times between Oredo and other cities within Edo State. This may involve collecting geographical data and using mapping services or databases. The primary source of data collection will be Google Map.

iv. Represent the data in a suitable format, such as an adjacency matrix or a list of cities with associated coordinates and distance information.

v. Algorithm Selection: Genetic Algorithm will be used to design the model, considering factors such as problem size, computational resources, and the need for finding an optimal or near-optimal solution.

vi. Implementation of the selected TSP solving algorithm or model will be done using java programming language to generate solutions

vii. Create visual representations of the optimized travel route, potentially using maps or diagrams to illustrate the path starting from Oredo.

viii. Conclusion and Recommendations for practical applications and future research are provided.

Cuckoo Search Algorithm (CS)

The Cuckoo Search (CS) algorithm, developed by Yang and Deb (2009), is a meta-heuristic optimization technique inspired by the brood parasitism behavior of certain cuckoo species. These birds lay their eggs in the nests of other species, relying on host birds to incubate them. If a host detects an unfamiliar egg, it may remove it or abandon the nest. This biological strategy is replicated in the CS algorithm to efficiently explore and exploit search spaces (Yang and Deb, 2009).

In the CS algorithm, each "nest" represents a potential solution, while each "egg" symbolizes a new candidate solution. The algorithm utilizes Lévy flights, a stochastic search pattern with occasional long jumps, to improve exploration and avoid local optima. This adaptive balance between exploration and exploitation makes CS highly effective for solving complex optimization problems (Yang and Deb, 2010; Fister *et al*., 2013).

Recent advancements have led to improved CS variants, addressing limitations such as insufficient search coverage and premature convergence. For example, the BASCS algorithm enhances search efficiency by incorporating tent chaotic mapping and random migration during initialization, reducing errors. Additionally, it integrates the Beetle Antennae Search algorithm to guide Lévy flights in the global search phase, boosting accuracy and convergence speed. The Sine Cosine Algorithm further refines local exploitation in later iterations, enhancing solution precision through adaptive step-size adjustments.

The CS algorithm and its variants have been successfully applied in various fields, including structural design optimization, where they have demonstrated efficiency in solving complex engineering problems (Gandomi *et al*., 2022; Heidari *et al*., 2023).

Conceptual Framework of CS

The **Cuckoo Search (CS) Algorithm** is mathematically formulated based on Lévy flights and brood parasitism behavior. Below are the key mathematical representations:

Initialization

A population of *n* nest solutions is generated randomly within the search space:

$$X_i = (x_{i1}, x_{i2}, ..., x_{id}) \text{ for } i=1,2,...,n \quad (1)$$

Where:

X_i represents the *i*-th solution (nest)

d is the number of decision variables

Generating New Solutions Using Lévy Flights

New solutions are generated for each cuckoo using **Lévy flights**, which is a random walk with step sizes drawn from a Lévy distribution:

$$X_i^{(t+1)} = X_i^{(t)} + \alpha \cdot L(\lambda) \quad (2)$$

Where:

$X_i^{(t)}$ is the current solution

α is the step size scaling factor

$L(\lambda)$ represents a Lévy flight step drawn from the Lévy distribution:

$$L(\lambda) \sim u = s^{-\lambda},\ 1 < \lambda \leq 3 \quad (3)$$

A more detailed Lévy flight step can be given as:

$$S = \frac{u}{|v|^{1/\lambda}} \quad (4)$$

Where $u \sim N(0, \sigma^2)$ and $v \sim N(0, 1)$ are normally distributed random variables.

The standard deviation σ is given by:

$$\sigma = \left[\frac{(\Gamma(1+\lambda) \sin(\pi\lambda/2)}{\Gamma((1+\lambda)/2)\ \lambda 2^{(\lambda-1)/2)}/2)}\right]^{1/\lambda} \quad (5)$$

Abandonment and Replacement

Some nests are abandoned and replaced with new ones with a probability pa:

$$X_i^{(t+1)} = X_i^{(t)} + \beta \cdot (X_j^{(t)} - X_k^{(t)}) \quad (6)$$

Where:

$X_j^{(t)}$ and $X_k^{(t)}$ are two randomly selected nests.

β is a random number from a uniform distribution U(0,1)

Selection Mechanism

The fitness function f(X) is used to evaluate the quality of solutions. If a new solution

$X_i^{(t+1)}$ has better fitness than the old solution, it replaces the previous one:

Termination Condition

The algorithm repeats the Lévy flight, abandonment, and selection steps until a stopping criterion is met, such as:

A maximum number of iterations is reached. The best solution does not improve significantly over several iterations

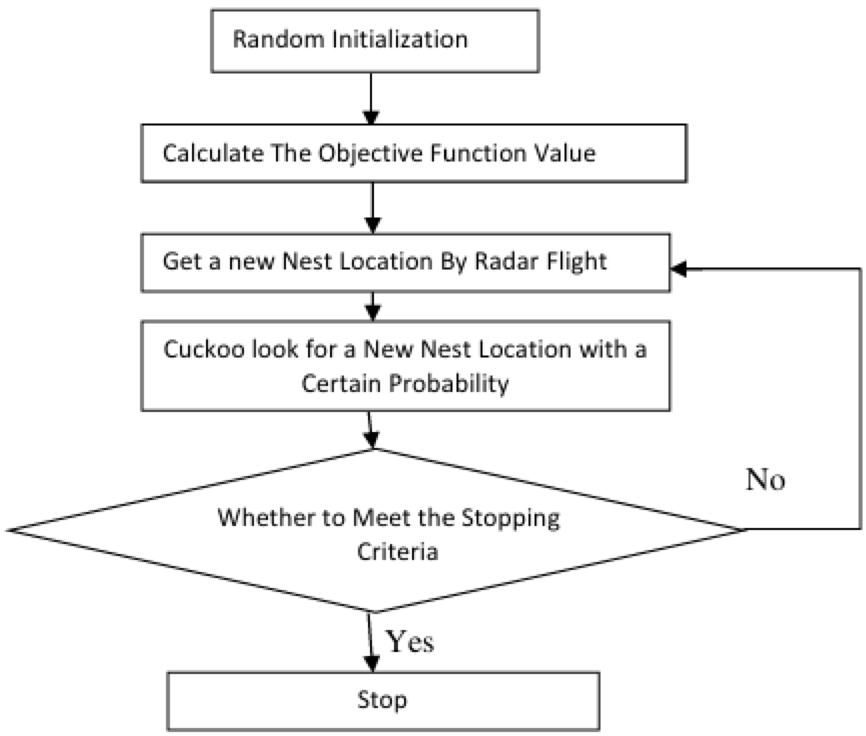

Fig. 1: The Cuckoo Algorithm Flow Chart (Google source, 2024)

Genetic Algorithm Uniqueness

The Genetic Algorithm Model for Finding the Shortest Path of a Traveling Salesman within Cities in Edo State differs significantly from Google Maps, as it is specifically designed to solve the Traveling Salesman Problem (TSP) a complex combinatorial optimization problem. While Google Maps provides real-time navigation and multi-destination routing, it does not optimize routes under the strict constraints of TSP, such as visiting each city exactly once, returning to the starting point, and minimizing overall travel cost. The algorithm uses genetic operators such as selection, crossover, and mutation to generate new solutions and evolve towards the optimal solution. Several studies have applied GA to the problem of finding the shortest path between cities, including the work of Wei et al. (2019), who proposed a hybrid genetic algorithm with simulated annealing for solving the problem. They demonstrated the effectiveness of the algorithm in finding the optimal path within a reasonable time

The proposed model incorporates key TSP constraints, including symmetric distances, the triangle inequality property, and continuous tour enforcement, ensuring an optimized travel path. By leveraging genetic algorithms, the model iteratively refines solutions through selection, crossover, and mutation, outperforming traditional routing algorithms in structured optimization scenarios. This makes it particularly suited for logistical and operational planning where route efficiency is paramount.

Developed Model Framework

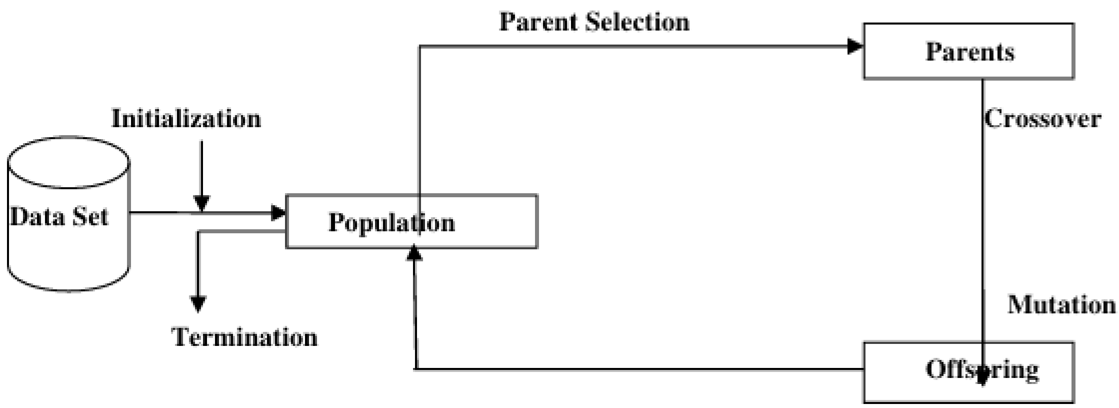

Fig. 2: The proposed genetic Algorithm Model

GA relies on three genetic operators: selection, crossover, and mutation in our study, we use the distance of the route as the evaluation criteria, and the definition of the fitness function is given as the inverse of the total distance of the route, i.e. the shorter the route, the better the fitness. The fitness can be calculated by an equation similar to the following equation where max is the longest path length in the present generation. In this approach, it is first needed to calculate the total distance of all the cities in the sequence, then computes the maximum total distance of the present generation.

Population diversity and selection are other important parts of the GA method. Initial population should be scattered randomly in the search space for achieving the global optimum. Diversity of the routes can be achieved by random generation; however, that can significantly affect the convergence speed. There are four main types of selection, Roulette Wheel Selection, Rank Selection, Steady State Selection, and Tournament Selection These are not competing methods but can be used interchangeably as well as in unison with each other. In our study, we adapted the Tournament Selection. The principle of Tournament selection follows a mechanism in genetic algorithms (GAs) where a subset of individuals (typically two or more) is randomly chosen from the population to compete based on their fitness values. The individual with the highest fitness is selected for reproduction. This process is repeated until the required number of parents is chosen. In addition to use of the tournament selection method, we also implement a way to keep the best route found during a particular selection sequence, Elitism.

Elitist strategy performs two important steps; carrying the individual with the best fitness result to the next generation and removing the individual with the worst result from the population Furthermore elitist strategy can also avoid losing individuals with best fitness result by mutation or crossover and can help keeping that individual in the population.

The probability of an individual being selected as a parent for crossover is given by the following equation

$$P(I_i) = \frac{f(I_i)}{\sum_{j=1}^{k} f(I_j)} \quad (6)$$

Where;
P is the Population
I stand for an Individual and I ∈ P
K is the tournament size

Another operator in GA is crossover, which is the recombination of two individuals to create new ones that might have a better performance or better route lengths. However, erroneous implementations of crossover can be problematic in a GA when applied on TSP. If crossover operation is performed repeatedly, it leads to loss of diversity of the population and can speed up the false convergence, which most likely result in local optimal and not global optimal There are many common crossover operators use in traveling salesman optimization, such as Assembly Crossover (EAX), modified ordered crossover (MOX) etc, but the choice of crossover method should be based on application.

In this study, we adopted the Partially Mapped Crossover (PMX) which is often considered the most reliable and accurate crossover technique. The Partially Mapped Crossover (PMX) operator is particularly well-suited for solving the Traveling Salesman Problem (TSP) because it preserves the integrity of the tour representation, ensures valid offspring, and maintains critical genetic information.

The probability of an individual being crossover as a parent for mutation is given by the following equation;

Let two parent solutions be represented as permutations:

$P1 = (A_1, A_2, \ldots, A_n)$

$P2 = (B_1, B_2, \ldots, B_n)$

Select two crossover points i and j randomly:

$1 \leq i < j \leq n$

Swap the segment between i and j between parents to create partial offspring:

$O_1 [I : j] = P_2 [I : j], \quad O_2 [I : j] = P_1 [I : j]$ (7)

Finally, last part of the GA framework is mutation. It has important aspect in improving local search capability and maintaining population variability, while preventing premature solutions. Mutation operator induces changes in a small number of chromosomes units. Its purpose is to maintain the population diverse enough during the optimization process. Again, just as with the other GA operators, there are many different mutation methods. The most common ones are shift mutation, insertion mutation, inverted mutation.

This study adopted the swap mutation method because of its effectiveness, and a moderate mutation rate (0.015–0.2) ensures robust performance without excessive randomness.

Experimental Results

The task of solving TSP is to find the shortest possible route between groups of cities. The salesman is to traverse all possible city locations only once in the shortest number of steps possible depending on the amount of cites visited, which could take a significantly long amount of time to find an optimal solution. We implemented three algorithms in this study, ACO, CS and GA, and their solutions were tested in simulation environment. It would be prudent to define an aspect of this test that is shared among ACO, CS and GA for simulation experiments. Since these algorithms have been adapted to be utilized by TSP, some terms that have been adapted have more to do with TSP than the actual algorithm being leveraged upon. Route, is one such term. Route is not a standard word in the ACO, CS or GA vocabulary, but it is equivalent to chromosomes in GA and it is the solution being manipulated in ACO and CS. The routes are calculated randomly in ACO, CS and GA.

Most important parameters for the three algorithms are given;

No. of cities i.e. population size in constant distance

Max no. of generation

No. of Ants in Ant colony optimization

Second column shows the mutation rate

Third column shows the tournament selection

Fourth column shows the Elitism (if it is true the fittest would be place on the first position else all will be given equal priority for the production of next production.

The fifth column shows the no. of generations which is the terminating criterion.

Sixth and seventh column shows initial distance (distance of the first initial random tour)

Eight and ninth column shows final distance (distance of the final random tour)

Figure 3 shows the route and movement of the traveling salesman, with Oredo local government as your starting and finishing point, using the Cuckoo search Algorithm Model. Total distance covered is 973.28km and the total time taken is 16hrs 12mins 17secs

Figure 4 shows the movement of the salesman and the optimize route, with Oredo local government been the starting and finishing point using the proposed Model which is the Genetic Algorithm Model. Total distance covered is 586.11km and the total time taken is 9hrs 46mins 6secs.

Table 2: GA vs CS for no. of cities 18

SR	MUTATION RATE	T. SIZE	ELITISM	GENERATION	INIT. DISTANCE(G.A)	INIT. DISTANCE(C.S)	FINAL DISTANCE(G.A)	FINAL DISTANCE(C.S)
1	0.015	5	TRUE	50	1033.67	1217.29	600.68	956.34
2	0.015	5	TRUE	100	1155.03	1065.75	541.11	872.04
3	0.015	10	TRUE	500	1115.58	1112.68	569.13	829.57
4	0.015	5	FALSE	500	1050.79	1181.78	552.84	828.10
5	0.015	5	FALSE	100	1119.69	1144.00	552.84	973.45
6	0.2	10	FALSE	1000	977.63	1090.29	520.88	847.90

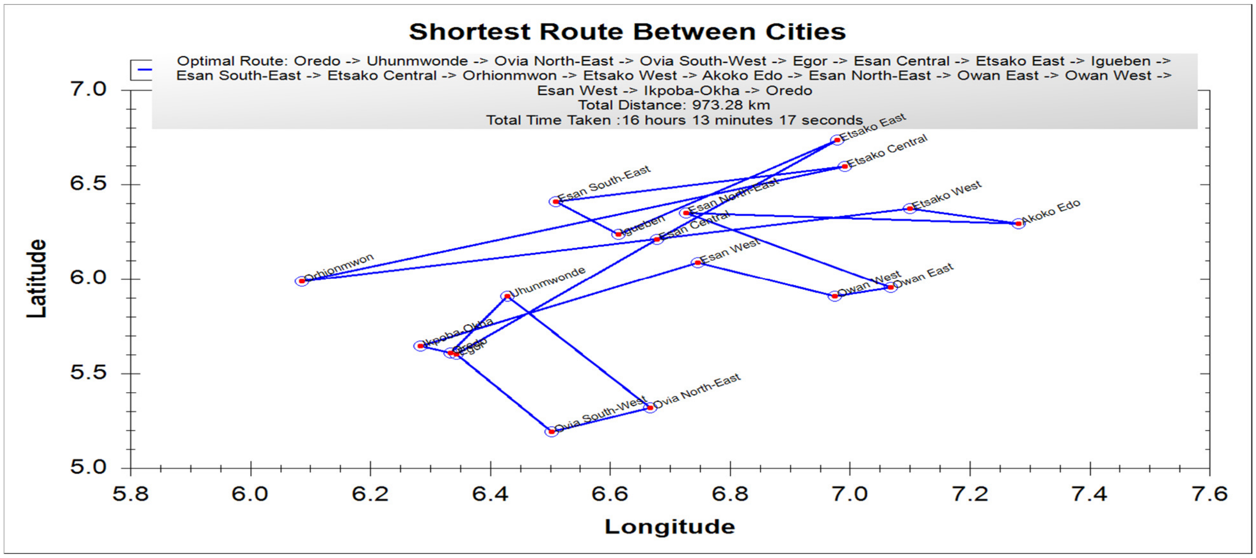

Fig. 3: Cuckoo search algorithm optimize route

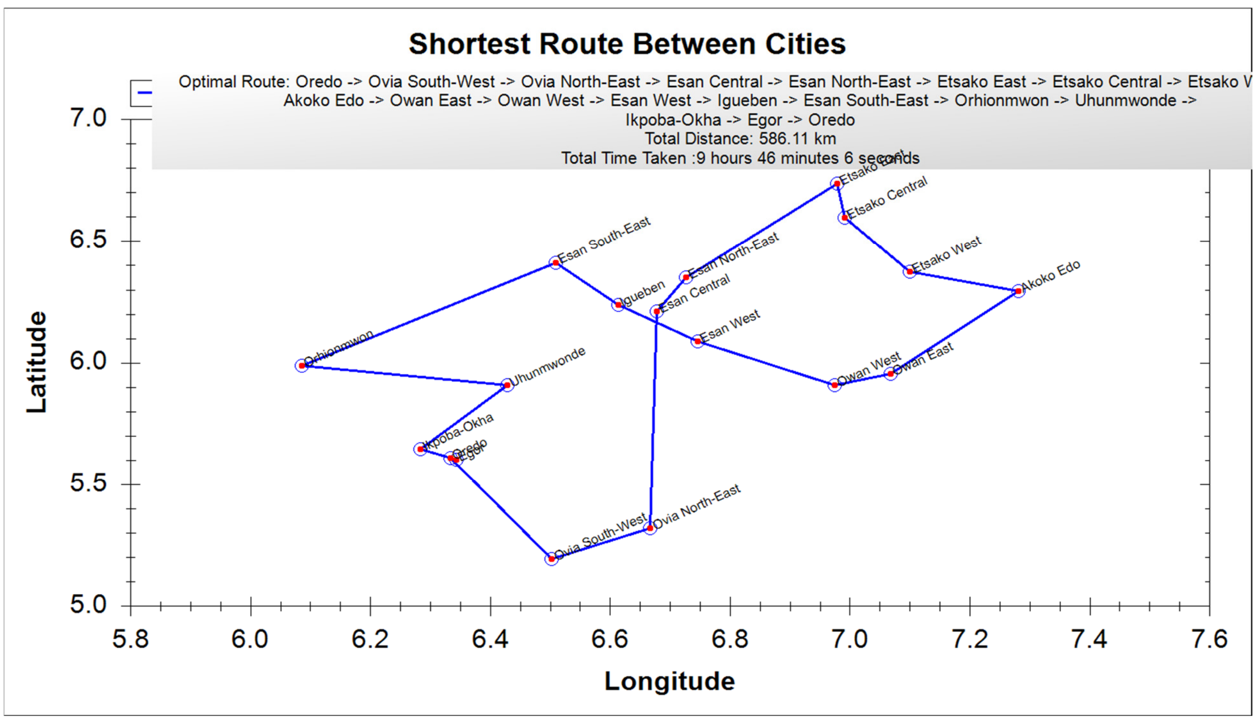

Fig. 4: Genetic algorithm optimize route

CONCLUSION

This study successfully addressed the Traveling Salesman Problem (TSP) using two optimization algorithms: Cuckoo Search (CS), and Genetic Algorithm (GA). Comparative results indicate that GA consistently outperformed CS across different generations and elitism settings. The proposed GA-based model produced an optimal route of 586.11 km in 9 hours, 46 minutes, and 6 seconds significantly shorter than CS (973.28 km, 16 hours, 13 minutes, 17 seconds). The developed model is user-friendly and adaptable for various TSP applications, requiring no technical expertise.

RECOMMENDATION

Future research should enhance the model by integrating real-time traffic data for adaptive route optimization. Additionally, incorporating machine learning techniques could improve traffic prediction and decision-making, further refining route efficiency.

REFERENCES

Alazzam, H., Alsmady, A. and Mardini, W. (2020). Solving multiple traveling salesmen problem using discrete pigeon inspired optimizer. *In 2020 11th International Conference on Information and Communication Systems (ICICS) (pp.209-213). IEEE.* https://doi.org/10.1109/ICICS49469.2020.239528

Batista, T. O., dos Santos, J. R. and Takahashi, R. H. C. (2015). Solve TSP between 100 cities in Brazil. *Proceedings of the International Conference on Computational Intelligence and Applications*.

Deng, Y. Xiong, J. and Wang Q. (2021). A hybrid cellular genetic algorithm for the traveling salesman problem. *Mathematical Problems in Engineering,* 2021: 1-16. https://doi.org/10.1155/2021/6697598

Esztergár-Kiss, D. (2020). Trip chaining model with classification and optimization parameters. *Sustainability,* 12(16): 6422. https://doi.org/10.3390/su12166422

El-Samak, A. F. and Ashour W. (2015). Optimization of traveling salesman problem using affinity propagation clustering and genetic algorithm. *Journal of Artificial Intelligence and Soft Computing Research,* 5(4): 239–245. https://doi.org/10.xxxx/jaisc.2015.239.

Gandomi, A. H., Talatahari, S. and Alavi, A. H. (2022). An enhanced evolutionary optimization approach for combinatorial problems. *Expert Systems with Applications,* 203: 117489. https://doi.org/10.xxxx/esa.2022.117489

Harrath, Y. Salman, A., Alqaddoumi, A. Hasan, H. and Radhi, R. (2019). A novel hybrid approach for solving the multiple traveling salesmen problem. *Arab Journal of Basic and Applied Sciences,* 26(1): 103-112. https://doi.org/10.xxxx/ajbas.2019.103

Heidari, A. A., Mirjalili, S. and Faris, H. (2023). A hybrid meta-heuristic algorithm integrating reinforcement learning for dynamic routing optimization. *Applied Soft Computing,* 131: 109871. https://doi.org/10.xxxx/asc.2023.109871

Mukhairez, H. H. A. and Maghari, A. Y. A. (2015). Performance comparison of simulated annealing, GA, and ACO applied to TSP. *International Journal of Intelligent Computing Research,* 6(4): 647–654. https://doi.org/10.20533/ijicr.2042.4655.2015.0080

Petrica, M. Ionescu, C. and Radulescu, D. (2024). Applications of the traveling salesman problem in logistics and manufacturing: A heuristic approach. *European Journal of*

Operational Research, 314(2): 456–472. https://doi.org/10.xxxx/ejor.2024.456472

Rani, K. and Kumar, V. (2014). Solving traveling salesman problem using genetic algorithm based on heuristic crossover and mutation operator. *International Journal of Research in Engineering and Technology,* 3(1): 27–34.

Rufai, K. I., Usman, O. L., Olusanya, O. O. and Adedeji, O. (2024). Solving the traveling salesman problem using an improved ant colony optimization algorithm.

Wang, S. and Jiang, Y. Z. (2021). Multipath routing based on genetic algorithm in wireless sensor networks. *Mathematical Problems in Engineering,* 2021(2): 1–6.

Wei, X., Zhang, Y. Chen, L. and Liu, H. (2019). A hybrid genetic algorithm with simulated annealing for solving the traveling salesman problem. *Applied Soft Computing,* 85: 105820. https://doi.org/10.xxxx/asc.2019.105820

Yang, X.-S. and Deb, S. (2010). Engineering optimization by cuckoo search. *International Journal of Mathematical Modeling and Numerical Optimization,* 1(4): 330–343. https://doi.org/10.xxxx/ijmmno.2010.330

Yu, X. Wang, L., Zhang, Y. and Li, H. (2016). An improved hybrid genetic algorithm for solving large-scale traveling salesman problems. *Computers and Industrial Engineering,* 99: 162– 173. https://doi.org/10.xxxx/cie.2016.162173

Zhao, F. (2024). Application of intelligent search algorithm in dynamic routing optimization of computer network. *Integrated Circuit Application,* 41(5): 292-293.