

A DECISION SUPPORT SYSTEM BASED ON OPTIMIZATION METHODS FOR OPTIMIZING DUHOK UNIVERSITY TRAFFIC PROBLEM

SAGVAN ALI SALEH*, SERWAN ALI MOHAMMAD*, RAFID SALIH SARHAN*,
RAWAN RASHAD KAMAL**, HUDA MAWLOUD AHMED**, AHLAM YOUSIF YOUNIS**, and
KAHRAMAN KHODEDA MIRZA**

*Dept. of Electrical and Computer Engineering, University of Duhok, Kurdistan Region–Iraq

**Student in Electrical and Computer Engineering, University of Duhok, Kurdistan Region–Iraq

(Accepted for Publication: November 27, 2023)

ABSTRACT:

Combinatorial optimization problems have previously proven to be significant and appealing for both basic research and practical applications. In this work, a real-life problem at Duhok University is considered, named as Duhok University Traffic Problem. Traffic bottlenecks, difficulties in moving, transportation cost in terms of time and financial and other problems in Duhok University are addressed as a series problems. In this paper a decision support system based on optimization methods are proposed in order to optimize a public transportation network to reduce the impact of the considered problem. The authors are first simulated the problem as a well-known optimization problem named as travelling salesman problem. Second, the problem is represented mathematically by introducing a mathematical programming model. Third and last, a decision support system based on three optimization methods are proposed in order to help in solving the considered problem, which are: greedy method, local search method, and artificial bee colony method. Computational results showed that, all the three proposed algorithms optimized and enhanced the transportation network in Duhok University, where Artificial Bee Colony presented high quality improvement by 36.48% and outperformed both, the greedy algorithm (13.81%) and the local search algorithms (31.45%).

KEYWORDS: Travelling Salesman Problem, greedy method, Artificial Bee Colony metaheuristic.

1. INTRODUCTION

Combinatorial optimization problems have already demonstrated their importance and attraction for both fundamental research and real-world applications. Indeed, many real-world situations have been treated and modeled as combinatorial optimization problems. However, a problem with transportation at Duhok University has been identified as a serious problem. With almost 26000 students and employers, several departments, and institutions, vast area, there are often severe traffic bottlenecks that affect both automobiles and pedestrians. That is, Duhok University Traffic Problem (abbreviated as DUTP). In order to address this problem, it became necessary to establish a public transportation network at Duhok University. Thus, the majority of people will use these public transportation services, reducing the flow of both vehicles and pedestrians,

and so reducing the impact of DUTP. The challenge is determining the shortest route between the

Stations, so that all buses visit every station in acceptable time.

However, DUTP can be simulated as a combinatorial optimization problem: Travelling Salesman Problem (TSP) (Qamar, et al. 2021). TSP is a renowned challenge in the fields of computer science and mathematics, requiring the identification of the most efficient route that encompasses a given set of cities and eventually returns to the initial location. The TSP is a relevant problem in transportation planning because it can help optimize routes for vehicles such as delivery buses. By finding the shortest possible route that visits all the required destinations, transportation can save time, fuel costs, and reduce vehicle emissions. In fact, a diverse range of real-world scenarios can be represented as problems within the TSP category, spanning across different domains (Qamar, et al. 2021).

Let $G = (V, E)$ be a directed weighted graph, where V is the set of vertices representing the cities, where $V = 1, \dots, n$, n is the total number of cities. E is the set of directed edges representing the roads connecting the cities. Let $c(i, j)$ be the cost of traveling from city i to city j , for all $i, j \in V$. The

$$\begin{aligned} \text{minimize } & \sum_i^n \sum_j^n c(i, j) x(i, j) && \dots(1) \\ \text{s.t. } & \sum_{i=1, i \neq j}^n x(i, j) = 1 \quad \text{for } j = 1, \dots, n && \dots(2) \\ & \sum_{j=1, j \neq i}^n x(j, i) = 1 \quad \text{for } i = 1, \dots, n && \dots(3) \\ & \sum_{i \in Q} \sum_{j \neq i, j \in Q} x(i, j) \leq |Q| - 1 \quad \forall Q \subseteq \{1, \dots, n\}, |Q| \geq 2 && \dots(4) \end{aligned}$$

Where, $x(i, j)$ is a binary decision variable that takes value 1 if the edge (i, j) is in the Hamiltonian cycle, and 0 otherwise. In this integer linear programming, there are four equations. Equation (1) is the objective function, where the goal is to find the shortest possible route. The first two constraints (Equation (2) and (3)) ensure that each city is only visited once. Equation (4) is the last constraint (called as sub-tour elimination constraint) ensures no proper subset Q can form a sub-tour (i.e., not one route but several sub routes), so the result is a single tour rather than the union of several smaller tours..

The remaining sections of the paper are arranged as follows. Section two reviews some related works. The principle of the suggested approach for optimizing the DUTP is covered in section three. In section four, the performance of the proposed approach is evaluated on a real-world data instance that is derived from actual measurements, and analyzes the obtained results. The paper's contribution is finally summarized in section five.

2. RELATED WORKS

The DUTP presents a practical challenge that can be modeled as the TSP, as discussed in the preceding section (Section 1). Consequently, the solution methods applicable to problems falling under the TSP category are also applicable to the DUTP, (for more details, see Kellerer *et al.*, 2014).

Huna *et al.*, in (Huan, Linh-Trung *et al.* 2013) presented a solution method based on Ant Colony Optimization (ACO) algorithm to solve the TSP. the authors proposed new ACO-based algorithms that address limitations of previous algorithms, such as premature convergence and computational

objective of the DTSP is to find a Hamiltonian cycle (i.e., a cycle that visits every vertex exactly once) of minimum total cost. The integer linear programming of considered DTSP can be written as following (Dao *et al.* 2016):

complexity. The proposed algorithms use strategies like local search and hybridization with other algorithms to improve ACO's performance. The authors provide a comprehensive overview of ACO for the TSP and proposes new efficient algorithms that can be applied to other combinatorial optimization problems.

Dao *et al.*, in (Dao, Abhary *et al.* 2016) presented a genetic algorithm for solving the travelling sales man problem). They proposed an algorithm in which they employs a sub-tour based crossover operator and a city swapping mutation operator. Their proposed algorithm is a valuable contribution to the field of TSP solving for solving large scale instances.

Zhao and Xu in (Zhao and Xu 2019) presents a hybrid algorithm that combines two optimization techniques, genetic algorithm and ant colony optimization, to solve the TSP. They show that a mix of genetic algorithms and ant colony optimization may provide high-quality results. The research provides a valuable contribution to the development of optimization techniques for solving the TSP and provide a reference for applying his proposed algorithm (the HHO algorithm) to other combinatorial optimization problems.

Qamar *et al.*, in (Qamar, Tu *et al.* 2021) developed a hybrid algorithm for solving the TSP based on the Best-Worst Ant System (BWAS) and particle swarm optimization. The both metaheuristic algorithms are combined. So they suggested technique balances solution quality and processing time, making it suitable for real-world applications.

Ilin (Ilin, Simić *et al.* 2022) proposed algorithm based on genetic algorithm for tour construction and tour improvement, and list-based simulated annealing. It showed a great behavior in their

solution quality and computational efficiency compared to the literature.

In this study, a decision support system is proposed to provide approximate solutions for optimizing DUTP using three optimization methods. Firstly, a greedy method is employed to generate a fast feasible solution. Secondly, a local search method is utilized as an intensification technique to refine the current solution. Lastly, the artificial bee colony method is employed as a diversification procedure to identify the best local optimum solution within the solution subspaces.

3. A DECISION SUPPORT SYSTEM FOR OPTIMIZING DUTP:

This study is dedicated to addressing the DUTP, a real-world issue directly linked to Duhok University. The authors possess an in-depth awareness of the substantial repercussions stemming from traffic congestion, delays experienced by individuals in reaching their destinations on time, the complexities of navigating the campus, and the various related challenges.

Nevertheless, this research relies on real data collected directly from the campus. The proposed decision support system incorporates three optimization techniques: a greedy approach, a local search strategy, and the ABC method.

3.1 Greedy method to produce a starting solution

The greedy method is a frequently used algorithmic approach recognized for its simplicity and effectiveness in addressing optimization challenges (Hifi, Saleh et al. 2015). It works by selecting the best local choices at each stage of problem-solving, without taking into account the overall global optimal solution. The greedy approach prioritizes immediate advantages and picks the most advantageous option at each decision point. Although the greedy method may not always guarantee the globally optimal solution, it provides a fast feasible solution. Its simplicity and speed make it suitable for problems to build a starting solution (Hifi, Saleh et al. 2014).

In the context of this study, the greedy method is one of the techniques utilized in the proposed decision support system to address the DUTP. It starts with a station and iteratively makes decisions that appear to be the best at that particular moment to build a feasible solution. It does not backtrack or reconsider previous choices once a decision is made. This characteristic makes the greedy method computationally efficient, as it avoids extensive exploration of all possible solutions, so can produce a fast solution.

Algorithm(1): -A greedy procedure for DUTP

Input	An instance of DUTP.
Output	A starting solution x .
1:	Select an arbitrary station as the starting point for the tour;
2:	Mark the starting station as visited;
3:	While there are unvisited stations remaining;
4:	Calculate the distance from the current station to all unvisited stations.
5:	Choose the nearest unvisited station as the next station to visit.
6:	Add the chosen station to the tour.
7:	Mark the chosen station as visited.
8:	Update the current station to be the chosen station.
9:	Once all stations have been visited, return to the starting station to complete the tour
10:	Calculate the total distance of the tour.
11:	Return the tour and its total distance;

Algorithm 1 illustrates the main steps of the proposed greedy procedure. We, recall that, this method is a fast, where, it can build a fast feasible solution of moderate quality.

However, the algorithm starts with an arbitrary station as the starting point. At each step, it selects the nearest unvisited station and adds it to the tour. This process continues until all stations are visited.

Finally, the algorithm connects the last station back to the starting point to complete the tour.

3.2 Local search method to improve the current solution

The local search method is a heuristic optimization technique used to find good solutions to complex problems (Hifi, Saleh et al. 2015). It operates by iteratively improving an initial solution through a process of neighborhood exploration.

Unlike systematic search methods that explore the entire search space, local search focuses on a smaller subset of neighboring solutions. The key idea behind the local search method is to exploit the local structure of the problem, gradually moving towards better solutions within the local neighborhood. It does not guarantee finding the global optimal solution but aims to find an acceptable solution within a feasible timeframe. Local search algorithms are widely used for

optimization problems where finding the exact optimal solution is computationally infeasible. They are particularly effective for NP-hard problems. By focusing on exploring nearby solutions, local search algorithms can efficiently navigate through the search space and converge to good-quality solutions (Aarts and Lenstra 2018).

In this study, a local search method is one of the techniques employed in the proposed decision support system to enhance the solution at hand.

Algorithm(2):- a local search procedure for DUTP

Input	A starting solution obtained from Algorithm 1.
Output	An enhanced solution
1:	Set the starting solution as the initial solution;
2:	Repeat the following steps until a stopping criterion is met (a maximum number of swapping):
3:	Generate a neighbor solution by applying local modifications to the current solution (swapping two stations).
4:	Calculate the quality of the neighbor solution using the objective function.
5:	If the best neighboring solution is better than the current solution, update the current solution to be the selected neighboring solution.
6:	Return the best solution found during the search process;

Algorithm 2 illustrates the main steps of the proposed local search procedure. The algorithm starts with an arbitrary station as the starting point. It begins with an initial solution and examines neighboring solutions by making small modifications to the current solution. It evaluates the quality of each neighbor using an objective function and moves to the neighbor that improves the solution. This process continues until no further improvements can be made, indicating a local optimum.

The local search algorithm iteratively explores neighboring solutions, selecting the best improvements until it reaches a local optimum. It explores the search space in a localized manner, focusing on improving the solution within the immediate neighborhood (Aarts and Lenstra 2018). However, it's important to note that the local search algorithm does not guarantee finding the global optimal solution for DUTP, as it can get stuck in local optima.

3.3 Artificial Bee Colony method to produce best local optimum

The Artificial Bee Colony (ABC) is a metaheuristic algorithm inspired by honey bee foraging behavior. Introduced by Karaboga in 2005, ABC is a nature-inspired optimization algorithm that aims to solve complex combinatorial optimization problems by imitating the collective

intelligence and cooperation observed in a bee colony (Blokdyk 2018).

By utilizing a population of artificial bees, the ABC algorithm iteratively refines and improves candidate solutions. The algorithm consists of employed bees, onlooker bees, and scout bees. Employed bees explore the search space by improving their solutions through local search, while onlooker bees evaluate the quality of solutions found by employed bees and choose promising solutions to further explore. Scout bees play a role in exploring unexplored regions of the search space. The ABC algorithm strikes a balance between exploration and exploitation, allowing it to efficiently navigate the solution space and converge towards optimal or near-optimal solutions. It offers flexibility in handling various types of optimization problems, making it a versatile algorithm for a wide range of applications (Cruz, González et al. 2010). The ABC method consists of three main phases: initialization, employed bee phase, and onlooker bee phase:

- The initialization phase creates an initial population of solutions randomly. The population size is typically set to a fixed value.
- In the employed bee phase, each solution in the population is improved by a local search. The local search operator is applied to each solution in the population to generate a new solution. The fitness

of the new solution is evaluated, and if it is better than the current solution, it replaces the current solution. The best solution found in this phase is selected as the reference solution.

- In the onlooker bee phase, the solutions are evaluated based on their fitness, and the best solutions are selected for the next generation. The

probability of a solution being selected is proportional to its fitness value.

The ABC algorithm is tailored to address the DUTP. It can help minimize transportation costs and make significant progress in mitigating traffic-related challenges, enhancing accessibility, and optimizing transportation within Duhok University.

Algorithm (3):- ABC to reproduce a series of evolved solutions

Input	"Pop" the population obtained from Algorithm 2.
Output	S_{best} : best solution.
1:	Initialization
2:	Determine the number of employed bees (N) and the number of food sources (SN) to represent potential solutions
3:	Generate an initial population of random solutions (food sources) representing tours for the TSP
4:	Evaluate the fitness of each food source using an objective function (total distance or cost of the tour)
5:	Employed Bee Phase: For each employed bee i ($1 \leq i \leq N$)
6:	Choose a neighbor solution (food source) j randomly, different from the current solution i
7:	Perform a local search around solution i to generate a new solution i' by applying a neighborhood exploration method specific to TSP (2-opt swap)
8:	Evaluate the fitness of the new solution i' by calculating the total distance or cost of the tour
9:	Onlooker Bee Phase:
10:	Calculate the probabilities of onlooker bees to choose a food source based on their fitness values
11:	For each onlooker bee k ($1 \leq k \leq N$)
12:	Select a food source i based on the calculated probabilities
13:	Perform a local search around solution i to generate a new solution i' using a neighborhood exploration method
14:	Evaluate the fitness of the new solution i' by calculating the total distance or cost of the tour
15:	Scout Bee Phase
16:	Identify the food sources that have not been improved for a specified number of iterations (limit)
17:	Replace the unimproved food sources with new random solutions (food sources) to encourage exploration
18:	Memorize Best Solution
19:	Keep track of the best solution (food source) found so far, which represents the tour with the minimum total distance or cost
20:	Repeat Steps 2 to 5 until a stopping criterion is met (a maximum number of iterations)
21:	Return the best solution found, which represents the optimal or near-optimal tour for the TSP

By following these steps, the ABC algorithm iteratively explores and refines potential solutions for the DUTP. The employed bees and onlooker bees perform local searches to improve the tours, while the scout bees introduce exploration by generating new random solutions. Through this iterative process, the algorithm aims to converge towards the tour with the minimum total distance or cost, providing an approximate solution to the DUTP of good quality.

4. COMPUTATIONAL RESULTS AND DISCUSSION:

In this section, the effectiveness of the proposed methods is evaluated on the real data instance collected from Duhok University. The instances consists of 13 stations, which are the main stations exist in the University. The proposed algorithms was coded in Python and tested on core i5 2.4 GHz.

We recall that, the focus is on route planning, aiming to enhance transportation and urban mobility in the area. This is achieved by optimize routes and present a transportation network with a

minimum distance. However, first step in the computational results evaluates the performance of the greedy method, as shown in Table 1.

Table(1):-Performance of the greedy algorithm.

	Actual distance	Distance achieved by greedy algorithm in 0.004 s	% of enhancement
Val.	14910 m	12850 m	13.81%

As shown in Table 1, the distance achieved by greedy algorithm is 12850m, i.e., with improvement of 13.81% of the actual transportation network distance in Duhok University.

In the second step of the computational results, the performance of the local search method is evaluated, as shown in Table 2.

Table (2):- Performance of the local search algorithm.

	Actual distance	Distance achieved by local search algorithm in 0.052 s	% of enhancement
Val.	14910 m	10220 m	31.45%

As illustrated in Table 2, the distance achieved by the proposed local search algorithm is 10220 m. The improvement is 31.45% when it is compared with the exist one.

In the third step of the computational results, the performance of the ABC algorithm is evaluated. Table 3 illustrates the best local optimum solution achieved in 100 iteration by the proposed ABC algorithm.

Table (3):- Performance of the ABC algorithm.

	Actual distance	Distance achieved by ABC algorithm in 2.546 s	% of enhancement
Val.	14910 m	9470 m	36.48%

Table 3 shows the distance achieved by ABC algorithm which is 9470 m, with improvement of 36.48% of the actual transportation network distance in Duhok University.

As it is cleared that, the solution quality achieved by the proposed ABC algorithm is better than those obtained by the greedy and local search algorithms, as illustrated in Table 4.

Table (4):- Performance of the ABC algorithm with compare with the greedy and local search algorithms.

Actual Transportation distance	Greedy Algorithm		Local search algorithm		ABC algorithm	
	Value	% of enhancement	Value	% of enhancement	Value	% of enhancement
14910 m	12850 m	13.81%	10220 m	31.45%	9470 m	36.48%
	0.004 seconds		0.052 seconds		2.546 seconds	

ABC algorithm yields solution distance of 9470 m in 2.546 seconds which represents the minimum transportation network distance can be achieved within the three proposed algorithms. Meanwhile, the greedy method yields solution of 12850 m in 0.004 second, and the local search method yields 10220 m in 0.052 second. Due to their membership in the metaheuristic family, where appropriate

solution space diversity is accomplished, the ABC algorithms outperform other algorithms. Meanwhile, there are no diversification techniques in both greedy and local search algorithms, which make the search within the solution space ineffective.

5. CONCLUSIONS

In this paper, we proposed a decision support algorithms for optimizing a real life problem in Duhok University, named as Duhok University Traffic Problem. The objective was to optimize routes, reduce travel times, and enhance the overall efficiency of transportation networks in Duhok University. First, this practical situation in real-life has been simulated as an optimization problem know as Travelling Salesman problem and a mathematical representation for the problem was presented. Then, three algorithms have been proposed as a decision-maker to help in optimizing the situation: greedy algorithm, local search algorithm, and Artificial Bee Colony. Computational results showed that, all the three proposed algorithms optimized and enhanced the transportation network in Duhok University, where Artificial Bee Colony presented high quality improvement by 36.48%, which outperformed the greedy and local search algorithms, where the improvement were (13.81%, and 31.45% respectively). As a result, it is advised to use the results from this article to solve the DUTP in order to enhance the road system at Duhok University.

REFERENCES:

- Aarts, E. and J. K. Lenstra (2018). Local Search in Combinatorial Optimization, Princeton University Press.
- Blokdyk, G. (2018). Artificial Bee Colony Algorithm: Creating Value, CreateSpace Independent Publishing Platform.
- Cruz, C., et al. (2010). Nature Inspired Cooperative Strategies for Optimization (NICSO 2010), Springer.
- Dao, S. D., et al. (2016). An effective genetic algorithm for large-scale traveling salesman problems. Proceedings of the World Congress on Engineering and Computer Science.
- Hifi, M., et al. (2014). A fast large neighborhood search for disjunctively constrained knapsack problems. Combinatorial Optimization: Third International Symposium, ISCO 2014, Lisbon, Portugal, March 5-7, 2014, Revised Selected Papers 3, Springer.
- Hifi, M., et al. (2015). "A hybrid guided neighborhood search for the disjunctively constrained knapsack problem." **2**(1): 1068969.
- Huan, H. X., et al. (2013). "Solving the traveling salesman problem with ant colony optimization: A revisit and new efficient algorithms." **2**(3-4).
- Ilin, V., et al. (2022). "A hybrid genetic algorithm, list-based simulated annealing algorithm, and different heuristic algorithms for travelling salesman problem."
- Qamar, M. S., et al. (2021). "Improvement of traveling salesman problem solution using hybrid algorithm based on best-worst ant system and particle swarm optimization." **11**(11): 4780.
- Zhao, P. and D. Xu (2019). Hybrid algorithm for solving traveling salesman problem. IOP Conference Series: Materials Science and Engineering, IOP Publishing.