Tuijin Jishu/Journal of Propulsion Technology
ISSN: 1001-4055
Vol. 44 No. 3 (2023)

A Comparative Case Study on Shortest Path of Environmentally Sustainable Multi-Objective Transportation Problem: Dijkstra's and Ant Colony Optimization Algorithms.

U. Modenisha, J. Merline Vinotha
PG & Research Department of Mathematics,
Holy Cross College (Autonomous),
Affiliated to Bharathidasan University,
Tiruchirappalli-620002,
India.

Abstract
Shortest path problem (SPP), is one of the most widely known topics in graph theory. Across various disciplines of SPP numerous real-world applications have been used. It is a combinatorial optimization (CO) problem that takes various forms. The main objective of this research is to determine a shortest path in a transport network based on a real-world case study that contemplates sustainable route planning. A comparative analysis of ACO and Dijkstra's algorithms are accomplished in this study. Based on the real-world scenario, we describe the single- and multi-objective ACO and Dijkstra's algorithm to determine the shortest path problem. We have incorporated a Python code to solve the second case.

Keywords: Shortest path problem, single-objective, multi-objective, Ant colony optimization, Dijkstra's Algorithm, Carbon emission.

Introduction:

In graph theory, "Shortest path problem" determines a path of minimum weighted edge between two vertices. In SPP road map can be modelled as a graph, wherein the intersection points represent the vertices and the road segments are represented by edges, which are assigned by weights. Ant Colony Optimization (ACO) is a metaheuristic algorithm. More precisely, it is an algorithmic framework that can be adapted to various problems like static and dynamic problems.

Marco Dorigo et al. (2004) published the first book on the Ant Colony Optimization (ACO) metaheuristic for solving combinatorial optimization problems. Marie Sawadogo et al. (2012) reduced the social and environmental impacts of intermodal transportation through path selection using a multi-objective shortest path approach. H. Fahmi et al. (2020). Solved the ACO for finding the nearest route in light food distribution. Ahuja, R. K. (1993), used the Dijkstra algorithm, to solve shortest path problems. S. Ubeda et al. (2011) determined a case study on green logistics at Eroski incorporating environmental management principles. Muhammad Adeel Javaid (2013) described a paper that provided a better understanding of the fundamental ideas behind Dijkstra. Amir Abbas Shojaie et al. (2021) discovered the shortest path in discrete-time dynamic networks based on the dual criteria of time and reliability by taking into account, the impact of delay times. Ghoseiri et al. (2010) proposed an algorithm based on multi-objective ant colony optimization (ACO). Cheng-Yuan Wu et al. (2018) considered both economic cost and environmental impact at once and extended the VRP into the GVRP by making the reduction of travel distance and carbon emissions its two objectives. D. Prayogo (2012) used ant colony behaviour to reduce construction costs and CO2 emissions. This paper states that the construction industry has become one of the major contributors to greenhouse gas emissions. Lu Chen et al. (2010) discovered the best route in a traffic network based on carbon emissions and addressed the problem of carbon emissions in transportation networks.
This paper incorporates carbon emissions for the first time and discusses about extended multi-objective pseudocode for shortest path problem.

The paper has been organized as follows: Section 2 deals with Preliminaries. Section 3 provides a methodology and formulations for both single and multi-objective ACO and Dijkstra's algorithm. Section 4 defines the case study and defines the problem. Section 5 provides the results of the case study, and a comparative study between the two algorithms are discussed.

2. Preliminaries:

2.1 Shortest path problem (SPP): "SPP" aims to minimize the sum of individual edges weights between two vertices in a graph.

2.2 Non – dominated paths: In a transportation network let P be set of all paths from node 0 to N, and let $g1(k)$= cost k, $g1(k)$= CO_2 emission (k), and $g1(k)$= reliability(k). A path $k1$ ∈P is said to dominate another path $k2$ ∈P if and only if the following conditions are fulfilled:

1. Path $k1$ is not worse than $k2$ in any objective (cost, emission, or reliability).
2. $k1$ outperforms $k2$ in at least one objective (cost, emission, or reliability), as shown in relation below:
3. $k1 \geq k2 \;\; iff \begin{cases} g1 + g2\;(k1) < g3(k2) \\ \quad g3(k1) > g3(k2) \end{cases}$

Path $k1$ does not dominate path $k2$ if any of the preceding conditions are violated. Among all paths P, the set of specific paths known as non-dominated paths, P_N, are those that are not dominated by any member of set P.

2.3 Ant colony optimization: Ant Colony Optimization (ACO) is a metaheuristic algorithm. The word "metaheuristic" consist of two Greek words. Heuristic derives from the verb heuriskein, which means "to find," and Meta on a contrast refers to "beyond, in an upper level." These are evolutionary algorithms that take their cues from nature. The first ACO algorithm was proposed by Marco Dorigo 1n 1992.

2.4 Dijkstra's algorithm: In 1956, A Dutch mathematician and computer scientist Edsger Wybe Dijkstra created this algorithm. It is a single-source shortest path algorithm. This algorithm is popular in determining the shortest.

3. Methodology:

The suggested study addresses a single-objective problem, creating a reliable logistics network that takes carbon emissions into consideration. The important objectives are to reduce carbon emissions and find the shortest path.

3.1 Notations:

C: cost for traversing each arc

E: carbon emission between each arc.

R: amount of reliability for each arc

$\Delta = (\eta, \varepsilon)$: graph with an ordered pair of vertex and edge

(x,y):an arc of the vertices x and y

$\mathcal{N}$: nodes in a graph

k: path of a directed weighted graph.

$\mathcal{R}o_x,o_{x+1}$: is the reliabilities assigned to each arc $(o_x,o_{x+1}) \epsilon\; k$

$\mathcal{C}o_x,o_{x+1}$: cost assigned to each arc $(o_x,o_{x+1}) \epsilon\; k$

Eo_x,o_{x+1} : emission assigned to each arc $(o_x,o_{x+1}) \epsilon\; k$

P^A_{xy}: Possibility of selecting the node y at the node x by A^{th} ant

μ_{xy} : Heuristic function or the attractiveness motion from node *x* to node *y*

υ_{xy} (t): Pheromone at t-time.

α: a constant amount that causes the pheromone concentration to intensify. The greater this amount is, the greater the importance of pheromone concentration and the less the effects of random behaviour become.

β: a constant amount for weighting motion attractiveness. The greater this value, the greater the effect of motion attractiveness.

ω: determines the number of ants.

N^A_x: The total selectable arcs for the A^{th}- ant located at the node x.

R': Corresponding reliability based on the bounds of the cost and emission criterion

Min_R **:** Minimum amount that should be taken into account for the reliability criterion.

Max_R **:** Maximum amount that should be taken into account for the reliability criterion

$Min_{(C+E)}$: The least amount taken into account for cost.

$Max_{(C+E)}$ **:** Maximum amount considered for cost in this problem

υ_{xy} : denotes the amount (intensity) of pheromone that is present on the arc (x,y)

γ : A pheromone evaporation constant value between the range zero and one.
D_{xy}: distance between arcs.
$\varepsilon(q)$: payload capacity of the vehicle.

δ : A constant value obtained by taking into account the remaining problem criteria, denotes the rate of general pheromone spilling.

Î : represents the Iteration process

$\emptyset$: Pheromone evaporation constant amount. The value lies between zero and one.

$\upsilon_0 = \frac{1}{n.Lnn}$ where the value of n is one and Lnn is the distance that has been traversed by the ant in an iteration.

$w_j = \text{Min}_i\ M_{ij}$

3.2 Case 1: Procedure for single-objective shortest path problem.
The workings of ACO can be informally described as follows:
Algorithm 1: ANT Colony based Optimization
Input: An instance x of a combinatorial optimization problem While the termination requirement is not satisfied.
Schedule activities
Ant based solution construction ( )
Pheromone update ( )
Daemon actions ( )
End scheduled activities
Sbest best solution among the possibilities that exist End while
Output: Sbest candidate to optimal solution for x

E_{xy} 's identification: E_{xy} represents the carbon emissions from node x to node y.

Initialization of parameters: The parameters follow predetermined values.

Inverse emission = $\mu_{xy} = \dfrac{1}{E_{xy}}$

Probability = $P_{xy}^{k}\left(t\right) = \dfrac{\left[\left(\upsilon_{xy}(t)\right)\alpha.\left(\mu_{xy}\right)\beta\right]}{\sum\left[\left(\upsilon_{xy}(t)\right)\alpha.\left(\mu_{xy}\right)\beta\right]}$

At each step, the ant will:

• select a point using maximum visibility measure or randomly.

• It saves the visited points in its memory so that it doesn't need to retrace the path.

Visibility measure (VM): $\sum\left[\upsilon(x,y)\right]\left[\mu(x',y')\right]\beta$

Select the node using visibility measure.
Make a note of the nodes that have been visited. Local pheromone update

$\upsilon(x, y) \leftarrow (1-\gamma).\ \upsilon(x', y') + \gamma.\ \upsilon_0$

Carbon (CO_2) emission: Ubeda et al. (2011) determined the emissions of carbon of a vehicle k traversing from customers x to y were calculated by $E_{xy} = D_{xy} \times \varepsilon(q)$.

Edsger W. Dijkstra, a Dutch computer scientist, came up with the idea for this algorithm in 1956. In a graph with weights, Dijkstra's algorithm determines the shortest route between a source node and all other nodes. The steps are as follows:

1. Set the carbon emission between the source and all other nodes
2. Mark the source node as having been visited.
3. Calculate the approximate carbon emission between each current node's unexplored neighbors and the current node. Update the distance if the guessed distance is shorter.
4. Select the unvisited node with the shortest speculative distance as the subsequent current node, and mark it as visited.
5. Continue performing steps 3 and 4 until you reach the final node.

3.3 Case 2:Methodology for Multi-objective SPP:

3.3.1 Procedure for multi-objective ACO

For any directed weighted graph with path $k = (o_1, o_2, \ldots o_x, o_{x+1}) \in \mathrm{P}$ the, cost, carbon emission and reliability are define as follows:

$$\text{Cost } (k) = \sum_{x=0}^{h} C_{o_x, o_{x+1}} \in \kappa \ \forall \kappa \in \mathrm{P} \tag{3.2.1.1}$$

$$\text{Carbon emission } (k) = \sum_{x=0}^{h} E_{o_x, o_{x+1}} \in \kappa \ \forall \kappa \in \mathrm{P} \tag{3.2.1.2}$$

$$\text{Reliability } (k) = \coprod_{(o_x, o_{x+1} \in \kappa)} \{R_{o_x, o_{x+1}}\} \ \forall \kappa \in \mathrm{P} \tag{3.2.1.3}$$

Eq. 3.2.1.4 is used to equalize the scales of these two criteria and to convert the reliability criterion's bounds to the cost criterion's ($\frac{1}{C+E}$) bounds because the bounds of these two criteria are not comparable.

$$R' = \frac{R - Min_R}{Max_R - Min_R} * \frac{Max_{(C+E)} - Min_{(C+E)}}{Max_{(C+E)} - Min_{(C+E)}} + \frac{1}{Max_{(C+E)}} \tag{3.2.1.4}$$

Eq. 3.2.1.5 shows the probability that an ant at node x will choose node y from a list of possible nodes while taking into account a network criterion known as motion attractiveness ($\mu_{xy} = \frac{Rxy}{Cxy}$). Instead of making a completely random choice, the ant in this instance is able to make a nearly intelligent choice.

$$P_{xy}^{o} = \begin{cases} \dfrac{\frac{Rxy}{Cxy + Exy}}{\sum_{x \in \mathrm{N}_x} \frac{Rxy}{Cxy + Exy}}, if \ \ y \in \mathrm{N}_x \\ 0 \qquad\qquad\qquad , otherwise \end{cases} \tag{3.2.1.5}$$

According to Eq. 3.2.1.6, each ant leaves some pheromone on the path after traveling from node x to node y.

$$\upsilon_{xy}^{new} = \upsilon_{xy}^{old} + \gamma . \upsilon_{xy}^{old} \tag{3.2.1.6}$$

Given the amount of pheromone present on the path, Eq. 3.2.1.7 illustrates the possibility of choosing node y for the ant located at node x of the existing paths.

$$P_{xy}^{\nu} = \begin{cases} \dfrac{\upsilon_{xy}}{\sum_{x \epsilon Nx} \upsilon_{xy}} \ \ if \ \ y \epsilon \ Nx \\ 0, \qquad otherwise \end{cases} \tag{3.2.1.7}$$

According to Eq. 3.2.1.8, the remaining pheromone on the route will dissipate

$$\upsilon_{xy}^{new} = \phi . \upsilon_{xy}^{old} \tag{3.2.1.8}$$

The final probability equation used by ants to choose the subsequent node is Eq. 3.2.1.9 below.

$$P_{xy}^{A} = \begin{cases} \dfrac{(\upsilon_{xy})^{\alpha}(\mu_{xy})^{\beta}}{\sum_{x \in \mathrm{N}_x^{a}} (\upsilon_{xy})^{\alpha}(\mu_{xy})^{\beta}}, if\ y \in \mathrm{N}_x^{A} \\ 0 \qquad , otherwise \end{cases} \tag{3.2.1.9}$$

The pheromone spilling update given by Eq. 3.2.1.10 is done by the ants to choose the optimal non-dominated paths. The amount of pheromone spilling will be higher for a better path. $\upsilon_{xy}^{new} = \upsilon_{xy}^{old} + \gamma . \upsilon_{xy}^{old} + \delta . \frac{Rxy}{Cxy}$

(3.2.1.10)

Pseudo-code of proposed multi-objective Ant colony algorithm:
Input: Consider a non-negative weighted graph *Δ= (ŋ,, ŧ).*
[C_{xy}, E_{xy}, R_{xy}],
An start ȷ ∈ ŋ , a termination ȴ ∈ ŋ.
Output: shortest paths from ȷ *to* ȴ
Initialization:
N: Node, Iteration, Ant, *α, β,γ,*∅
For *x*=1 to Iteration
For *y*=1 to Ant
Node=1
While Node< *N*
Determine the Heuristic Function
Choose the Next Node
Local Update Pheromone
End While
Crate Path by Ant *y*
Choose and Save Non-Dominant Paths in Iteration î
Global Update pheromone
End for î
Select Non-Dominate Paths in all Iteration

3.3.2 Dijkstra's algorithm for extended multi-objective:

A traditional bi-objective Dijkstra's algorithm was developed by computer scientist Edsger Dijkstra in 1956. To solve the SPP, a multi-objective Dijkstra's algorithm is developed.

The pseudo-code for extended multi-objective Dijkstra's algorithm is:

Input: consider a non-negative weighted graph *Δ= (ŋ, ŧ).*
{Cxy : (x, y) ∈ ŧ , Exy : (x, y) ∈ ŧ, Rxy : (x, y) ∈ ŧ }, an start ȷ ∈ *V* , a termination ȴ ∈ *V.*
Output: shortest paths from ȷ to ȴ
Initialization:
Ȥ: Three-dimensional Matrix [Cost, Carbon emission, Reliability], *ƙy=1*
For each child y of x do
Ȥ y[ƙy,1]= Ȥ x[ƙy,1]+C_{xy}
Ȥ y[ƙy,2]= $Ȥ_x$[ƙy,2]+E_{xy}
Ȥ y[ƙy,3]= $Ȥ_x$[ƙy,2]*R_{xy}
ƙy=ƙy+1
end for
Identify and Save Non-Dominated Paths for y (Optimal solutions)
if *y*= ȴ then

Tuijin Jishu/Journal of Propulsion Technology
ISSN: 1001-4055
Vol. 44 No. 3 (2023)

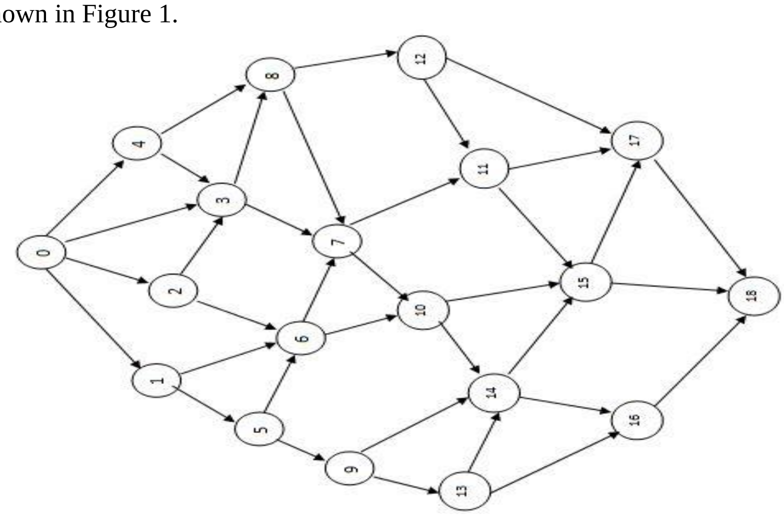
terminate
else
go to step 4
end if

4. Case study:

In various industries, efficient transportation networks are critical for optimizing logistics and resource allocation. This case study investigates the use of algorithms to find the shortest paths within a transportation network, emphasizing the importance of such optimization in lowering costs, reducing carbon emissions, improving time efficiency, and improving overall operational performance. A home appliance company uses a fleet of vehicles to deliver goods throughout the city. The task is to determine the most efficient routes for the vehicles to take in order to reduce carbon emissions while visiting the destination. The cargo vehicle being used is a diesel vehicle with a capacity of 3.5 tons. The current diesel price per litre in Pondicherry is approximately 86 rupees. The following are the vendors from whom orders are commonly placed: Kottakuppam(0), Kalapet(1), Ariyankuppam(2), Villianur(3), Tindivanam(4), Marakkanam(5), Thavalakuppam(6), Thirubhuvanai(7), Gingee(8), Alamparai(9), Manjakuppam(10), Villupuram(11), Thiruvannamalai(12), Mahabalipuram(13), Cuddalore(14), Chidambaram(15), Panruti(16), Sirkali(17), Karaikal(18). The graph representation numberings are provided for each places. The home appliances shop in Puducherry's Lawspet has its primary warehouse in Kottakuppam. This company has a legacy of five years and is currently expanding. It is a primary distributor that works with a variety of dealers to provide various versatile products. It is mainly known for the IFB and Samsung brands. In the distribution market, they have a few competitors. To sustain themselves among them, they have to plan their trips to dealers in an ingenious way. It is important to note that the distribution is done based on routes preferred by drivers, as they have a maximum time limit of 2 days to supply goods. The supply is done from 9 a.m. to 10 p.m. each day. On average, they get an order placed from at least one dealer per week in a month. Additionally, it is noted that during offer periods, they get the most drastic sales from all the dealers. They have a custom of advertising through pamphlets and audio casts every month. They preferably use one or a fleet of trucks for transport. To maintain their reputation and have stable, rapid growth, a nimble transport route has to be chosen.

4.1 Problem description:

The primary challenge in solving shortest path problems is to begin at a start node, traverse a series of middle nodes to reach the terminal node, and then optimize one or more criteria at decision points. Both discrete and continuous criteria can apply to these kinds of networks, depending on whether they are deterministic or stochastic. A transport network is created using the case study as a guide. The desired routes that are used for traversal were designed as a directed weighted graph. The total number of nodes in the graph, which represent each dealer in the home appliance store, is 19. The main goal of the problem is to travel from starting node to terminal node and find the shortest path out of all possible paths. The network designed to solve this issue is designated as $\Delta = (\eta, t)$, where "η" and "t" stand for the total number of nodes and the total number of directed arcs connecting those nodes, respectively. The edge weights of the graphs from node x to node y are carbon emission, cost, and reliability, which are represented by the symbols "*Exy*", "*Cxy*" and "*Rxy"*. The graphical representation is shown in Figure 1.



It is assumed that

1. The home appliance products have to be transported from Kottakuppam (the warehouse) to Karaikal through the network.
2. The movement will occur from start node 0 to terminal node 18.
3. The truck has a full payload capacity of 3.5 tons.

Case1: With regard to the transportation network, the single-objective approach uses Dijkstra's algorithm and conventional ant colony optimization to find the shortest path and minimize its carbon emissions.

Case2: The shortest path problem is solved using Python coding using a multi-objective approach that takes reliability, cost, and carbon emission criteria into account. From start 0 to terminal 18, the objective of this network is to simultaneously identify all non-dominated paths and the shortest path with the least amount of carbon emissions, the lowest cost, and the highest degree of reliability.

5 Results of the case study :

5.1 Case 1: Solution of single-objective shortest path problem.

5.1.1 Parameter initialization for ACO:

μ_{xy} (t)=0.1, α = 1, β =2, γ =0.5, ω = 4, $\mathcal{N}$=19.

The source and destination nodes are initially fixed, and the ants are allowed to find the shortest path by examining each path. In the first step**,** the ant travels from the source node to every possible and available node. The ant selects the best path based on the maximum visibility measure at each iteration. The visibility measure is computed by combining the pheromone level and inverse emission with the control parameters. The VM aids the ant's decision-making process. The pheromone value of the path traversed is then updated. After updating the pheromone value, the ant uses the visibility formula to select the next optimal path. It compares the values of two paths and then selects the highest value among them. It then updates the pheromone and repeats the process until the destination is reached. The ant remembers the path it took at each stage of construction. The path is developed in this manner at each stage or iteration. The process finishes when the ant arrives at its destination. Now, comparing all of the VM values and identifying the maximum value yields the optimal shortest path with the least CO_2 emission.

Iteration 1: VM = ∑ υ (x,y) μ (x`,y`) β
= (0.1)(0.02857)(2) + (0.1)(0.02857)(2) + (0.1)(0.02197)(2) + (0.1)(0.00649)(2)
= 0.01712

Pheromone update:
υ (0,1) = (1-0.5)(0.1) + (0.5*0.02857) = 0.064285
υ (0,2) = (1-0.5)(0.1) + (0.5*0.02857) = 0.064285
υ (0,3) = (1-0.5)(0.1) + (0.5*0.02197) = 0.060985
υ (0,4) = (1-0.5)(0.1) + (0.5*0.00649) =0.053245

Solving in similar manner we obtain the following solution.

The shortest path obtained by ACO is (0-2-6-10-14-16-18), which emits 598.5 grams of carbon.

5.1.2 Dijkstra's algorithm:

Initially Select the starting node on the defined transport network, visit it, and add the beginning node to the visited list. Following that, locate each starting node's immediate neighbour who has not yet been visited. Calculate and enter the emission for each of these neighbours. Select the next corresponding node with the smallest value. Check all the other possibilities from the starting node to all of the other nodes in the same pattern. Update the higher emission values with the lowest ones. Continue this process until the final node is reached from the initial node, traversing through the intermediate nodes with the lowest values. Once the terminal nodes are assigned a permanent value (i.e., the lowest emission value), the shortest path is identified and acquired.

Step	Visited nodes set	Unvisited node set	Paths
1	={0} (0)→(0)=0	={(1),(2),∞,(18)} 0-1= 35	0-1 = 35

Seek from (0)	0-2= 35 0-3= 45.5 0-4=154 $0 \rightarrow$others = ∞

Using a similar procedure, we arrive at the following solution.

The shortest path obtained by Dijkstra's algorithms is (0-2-6-10-15-17-18) with a least emission of 472.5 gram.

5.2 Case 2: Solution of multi-objective shortest path problem.

5.2.1 Multi-objective dijkstra's algorithm:

At the beginning of the first node, position the ants and initialize the parameters.
$\hat{I}$ = 100, ω = 8, α= 0.5, β= 0.9, γ = 0.1, $\emptyset$= 0.98, δ = 25 .
The ants are able to choose a better path using the probability formula. Each ant deposits some pheromone on the covered path after traveling from node x to node y. Some of the pheromone that is still present on the path will evaporate when an ant reaches the next node. The ant repeats this process until it reaches the last node. When a single ant or an entire colony arrives at the final node, pheromone spilling will update, and the best non-dominated paths will be determined.

5.2.2 Extended multi-objective dijkstra's algorithm:

The proposed extended Dijkstra multi-objective algorithm performs similarly to the single-objective model. In each stage of the single-objective Dijkstra's algorithm, the next node is chosen based on the minimal temporary labels of the preceding nodes, and this procedure is repeated for all the nodes. Extended Dijkstra, on the other hand, is based on the newly arrived nodes.

The shortest paths of both algorithms in Case 2 are determined by Python code. Among the non-dominated paths, the shortest paths obtained in ACO and Dijkstra's are represented in the in the table 1 are as follows:

Algorithm	Path	Cost	Carbon emission	reliability
ACO	[0, 2, 6, 10, 14, 15, 17, 18]	8844	483.0	0.07214062500000001
Dijkstra	(0, 2, 6, 10, 15, 17, 18)	8652	472.5	0.09112500000000001

Table1**:** represents shortest path among the non-dominated paths

6 Conclusion:

This paper, considers two cases that deal with single- and multi-objective transport network shortest path problems that mainly aim to reduce carbon emissions. In addition, it considers the cost and reliability in the second case. A comparative study has been made between both algorithms, and it can be concluded that the Dijkstra algorithm's shortest path yields a better outcome than the ACO algorithm. Dijkstra's shortest path is seen to be more effective in both the single-objective and multi-objective approaches to solving the transport network. Therefore, based on our case study, the home appliance store with a 3.5-ton transport truck can convey its home appliance products along the shortest route suggested by Dijkstra's method, enabling them to travel sustainably while using the least amount of fuel, time, and workforce. Their efficiency is increased, and this mode of transportation enables them to maintain and expand their market and helps them perform better than their rivals, which benefits both the distributors and their dealers. It predominantly provides a chance for an eco-friendly environment in Pondicherry, eventually enhancing global sustainability, which is an essential requirement in our world.

Acknowledgement:
This research work is a outcome of the programme M.Phil Mathematics. We gratefully acknowledged the same.

Reference:

[1] Ahuja, R. K. "Network flows (Doctoral dissertation, TECHNISCHE HOCHSCHULE DARMSTADT)." (1993).

[2] Akhtar, Aleem. "Evolution of Ant Colony Optimization Algorithm--A Brief Literature Review." arXiv preprint arXiv:1908.08007 (2019).

[3] Dorigo, Marco, Mauro Birattari, and Thomas Stutzle. "Ant colony optimization." IEEE computational intelligence magazine 1.4 (2006): 28-39.

[4] Fahmi, H., et al. "Ant Colony Optimization (ACO) Algorithm for Determining The Nearest Route Search in Distribution of Light Food Production." Journal of Physics: Conference Series. Vol. 1566. No. 1. IOP Publishing, 2020.

[5] Ghoseiri, Keivan, and Behnam Nadjari. "An ant colony optimization algorithm for the bi-objective shortest path problem." Applied soft computing 10.4 (2010): 1237-1246.

[6] Javaid, Adeel. "Understanding Dijkstra's algorithm." Available at SSRN 2340905 (2013).

[7] Prayogo, D., M. Y. Cheng, and D. K. Wibowo. "Minimizing Construction Cost and CO2 Emission Problem by Imitating the Behavior of Ant Colony." 1st International Conference on Sustainable Civil Engineering Structures and Construction Materials. 2012.

[8] Sawadogo, Marie, Didier Anciaux, and R. O. Y. Daniel. "Reducing intermodal transportation impacts on society and environment by path selection: a multiobjective shortest path approach." IFAC Proceedings Volumes 45.6 (2012): 505-513.

[9] Shojaie, Amir Abbas, and Seyed Esmail Seyedi Bariran. "A Comparison of Extended Dijkstra and ACO Algorithm for Shortest Path Problem in Dynamic Networks with Delay Times." *Advances in Industrial Engineering* 55.1 (2021): 1-26.

[10] Ubeda, Sergio, Francisco J. Arcelus, and Javier Faulin. "Green logistics at Eroski: A case study." International Journal of Production Economics 131.1 (2011): 44-51.

[11] Wu, Cheng-Yuan, Thammarsat Visutarrom, and Tsung-Che Chiang. "Green vehicle routing problem: the tradeoff between travel distance and carbon emissions." 2018 15th international conference on control, automation, robotics and vision (ICARCV). IEEE, 2018.

[12] Xiao-Yan, Liu, and Chen Yan-Li. "Application of Dijkstra algorithm in logistics distribution lines." Third International Symposium on Computer Science and Computational Technology (ISCSCT'10), Jiaozuo, PR China. 2010.