SAMUEL TIONG FU WEI

B.Sc. (Hons) STATISTICAL COMPUTING AND OPERATIONS RESEARCH

2024

OPTIMIZING CAPACITATED ELECTRIC VEHICLE ROUTE IN LOGISTIC OPERATIONS

SAMUEL TIONG FU WEI

BACHELOR OF SCIENCE (HONS) STATISTICAL COMPUTING AND OPERATIONS RESEARCH

FACULTY OF SCIENCE

UNIVERSITI TUNKU ABDUL RAHMAN

OCTOBER 2024

OPTIMIZING CAPACITATED ELECTRIC VEHICLE ROUTE IN LOGISTIC OPERATIONS

By

SAMUEL TIONG FU WEI

A project report submitted to the

Department of Physical and Mathematical Science

Faculty of Science

Universiti Tunku Abdul Rahman

in partial fulfilment of the requirements for the degree of

Bachelor of Science (Honours) Statistical Computing and Operations Research

October 2024

ABSTRACT

OPTIMIZING CAPACITATED ELECTRIC VEHICLE ROUTE IN LOGISTIC OPERATIONS

SAMUEL TIONG FU WEI

As the use of eco-friendly practices in logistics grows, optimizing routes for CEVs remains challenging due to their limited energy and load capacities. This project aims to develop an efficient route optimization algorithm for Capacitated Electric Vehicles (CEVs) in logistics, focusing on minimizing travel distance while adhering to vehicle capacity and battery limitations. To address this, the Ant Colony Optimization (ACO) technique was chosen for its ability to efficiently explore large solution spaces and identify optimal routes. The algorithm's performance was further enhanced by applying the Taguchi method to fine-tune key parameters to solve the Capacitated Electric Vehicle Routing Problem (CEVRP). Key parameters, such as the number of ants, pheromone influence, heuristic information, and evaporation rate, were optimized using the Taguchi method. The analysis showed that these parameters significantly impacted the route optimization, leading to a reduction in total travel distance. The results demonstrated that the optimized algorithm effectively minimized the distance traveled by CEVs while meeting operational constraints. This approach not only improves the efficiency of logistics operations but also contributes to sustainable transportation, making it applicable across various logistics and supply chain scenarios.

ACKNOWLEDGEMENT

Firstly, I would like to express my heartfelt gratitude to my supervisor, Pn. Nur Intan Liyana binti Mohd Azmi, for her invaluable guidance, support, and encouragement throughout the duration of this project. Her knowledge, suggestions, and guidance have been very helpful in determining the course and conclusion of this study. Without the appropriate guidance she gave me, I would not have been able to complete the writing for my final year project. Additionally, I would want to take this opportunity to express my gratitude to UTAR for providing me with the opportunity to conduct this study. Finally, I would like to thank my family for their unwavering support, understanding, and encouragement throughout my academic pursuits. Their love and encouragement have been my source of strength and motivation.

DECLARATION

I hereby declare that the project report is based on my original work except for quotations and citations which have been duly acknowledged. I also declare that it has not been previously or concurrently submitted for any other degree at UTAR or other institutions.

(SAMUEL TIONG FU WEI)

APPROVAL SHEET

This report entitled "**OPTIMIZING CAPACITATED ELECTRIC VEHICLE ROUTE IN LOGISTIC OPERATIONS**" was prepared by **SAMUEL TIONG FU WEI** and submitted as partial fulfillment of the requirements for the degree of Bachelor of Science (Hons) Statistical Computing and Operations Research at Universiti Tunku Abdul Rahman.

Approved by:

______________________ .

(MS. NUR INTAN LIYANA BINTI MOHD AZMI)

Date:…02/09/2024………

Supervisor

Department of Physical and Mathematical Science

Faculty of Science

Universiti Tunku Abdul Rahman

FACULTY OF SCIENCE

UNIVERSITI TUNKU ABDUL RAHMAN

Date: OCTOBER 2024

PERMISSION SHEET

It is hereby certified that **SAMUEL TIONG FU WEI** (ID No: **20ADB04885**) has completed this final year project entitled "**OPTIMIZING CAPACITATED ELECTRIC VEHICLE ROUTE IN LOGISTIC OPERATIONS**" under the supervision of MS. NUR INTAN LIYANA BINTI MOHD AZMI from the Department of Physical and Mathematical Science, Faculty of Science.

I hereby give permission to the University to upload the softcopy of my final year project in pdf format into the UTAR Institutional Repository, which may be made accessible to the UTAR community and public.

Yours truly,

(SAMUEL TIONG FU WEI)

TABLE OF CONTENT

LIST OF TABLES

LIST OF FIGURES

LIST OF NOTATIONS

$d_{depot,i}$	Distance from the depot to customer i
$d_{j,depot}$	Distance from customer j to the depot
d_{ij}	Distance between customers i and j
B	Maximum battery capacity of the EV
$T_{ij}(0)$	Initial pheromone level on the path between node i and node j
T_{ij}	Pheromone level on the path from node i to node j
n_{ij}	Heuristic information (inverse of distance)
α	Influence of pheromone
β	Influence of distance
ρ	Evaporation rate
Q	Pheromone deposit factor
y_i	Observed value of the response variable (total route distance) for the ith trial in this case
n	Number of trials or observations

LIST OF ABBREVIATIONS

ACO	Ant Colony Optimization
B&B	Branch and Bound
B&C	Branch and Cut
CEVs	Capacitated Electric Vehicles
CEVRP	Capacitated Electric Vehicle Routing Problem
DP	Dynamic Programming
EACO	Extended Ant Colony Optimization
EV	Electric Vehicle
EVs	Electric Vehicles
EVRP	Electric Vehicle Routing Problem\
MILP	Mixed-integer Linear Programming Models
MIP	Mixed Integer Program
PSO	Particle Swarm Optimization
SA	Simulated Annealing

CHAPTER 1

INTRODUCTION

1.1 Introduction

This research proposes to use capacitated electric vehicles (CEVs) in logistics operations to solve the challenging optimizing problem of routing these vehicles with an aim of improving the efficiency of the transports system. By using optimization methods, the goal of this project is to offer a study regarding the problems encountered in the CEV routing. It includes the research background, the problem or the research questions, the objectives of the research, the scope and the importance of the study.

1.2 Logistics Challenges in Sustainable Transportation

Transportation is one of the critical influencers of the performance of logistics in terms of the environmental impact of transportation services. In the past few years, there has been a rise in the demands for green or environmentally sustainable solutions such as the use of electric vehicles (EVs). All these vehicles are environmentally friendly through their capability to decrease carbon dioxide emissions, though their planning of routes is a little complex due to their restricted battery capacity and unique recharging demand. In the present work, the emphasis is made on the determination of the best routes available for the Capacitated Electric Vehicles to improve the logistics of the processes. The goal of this project is to facilitate an efficient low-cost transport system that can be used as a base to transport goods as the demand for the modern logistics increases.

Optimal CEV routing is required to minimize the distance that vehicles traverse, and energy consumed in delivering the required goods. On the other hand, this means that poor route planning may result in issues such as increased operational costs, longer time to the destination or even the dissatisfaction of the customers. In the logistics industry, efficient routing and scheduling of CEVs is significant especially when timely delivery of the goods is handy. However, current logistics practices may involve conventional routing techniques that might not capture all the complexities involved in the operationalization of CEVs, including constrained battery power, and necessity of recharging station. It can lead to inefficient routes where CEVs' full potential is not realised, and time and costs can be wasted. Further, the current systems might not be sufficient to effectively manage variability especially in the context of changing logistics requirements.

1.3 Challenges in Capacitated Electric Vehicles

As the logistics sector moves to adopting Capacitated Electric Vehicles (CEVs) to enhance sustainability and minimize greenhouse gases emissions, new issues emerge regarding the management of these vehicles in the network. CEVs bring numerous environmental advantages, however, they incorporate new difficulties in routing and logistics due to their short battery power and low load bearing capability. These factors greatly affect the logistics operation and call for proper planning of routes so that all customer needs are served through the available vehicles with optimal energy charge.

Thus, the ability of CEVs to travel a limited distance per charge means that their usage must be planned to ensure that the vehicle covers only the necessary distances in between charges. Also, integrated into the problem is the carrying capacity of the vehicle and here the CEVs have an extra constraint whereby they need to deliver all the required goods and not exceed their loading capacity. Overlooking any of these constraints results in determination of suboptimal routes, longer distances, higher operation costs and possibly longer delivery times.

The dynamic environment that is involved in the logistics operations where the customer requirements and the delivery points keep on changing, makes it even challenging to achieve these goals. Static scheduling and fixed route techniques which are commonly used are not effective in these situations mainly because they do not incorporate variations in capacity utilization and route choice.

This study seeks to overcome the above challenges by proposing a complex routing system for CEVs. This consider both the battery capacity that the vehicles have and the load-carrying capacity of the vehicles to plan the routes in such a way that all the customers' requirements are met. Thus, the research generates realistic demand scenarios and simulates the distance between the nodes to assess these routing strategies. The goal is to develop a solution where travel distance is minimized, and operating costs and overall sustainability of logistics operations are made efficient so that CEVs can be incorporated into new generation supply chains.

1.4 Problem Statement

Outdated route planning challenges are even more challenging to accommodate whenever one is factoring the range and battery power of electric cars. As the market shifts towards eco-friendly solutions, businesses require efficient delivery routes that reduce distances and energy expenditures while meeting customers' needs on time. The current solutions, however, are not sufficient to meet these needs by handling the constraints in real-time and determining the best routing, hence the need for a better and more adaptable algorithm (Gong et al., 2020).

1.5 Research Questions

The provided problem statement identifies several research challenges that this study aims to address. These challenges are explored through the following research questions:

I. In what ways could algorithm help in identifying and enhancing practical routes for capacitated electric vehicles (CEVs) in logistics?

II. How to evaluate the most efficient routings and recharging patterns that a vehicle can take to minimize total travel distance subject to the vehicle capacity and battery constraints?

1.6 Research Objectives

This study aims to optimize the routing of Capacitated Electric Vehicles (CEVs) in logistics operations to minimize total travel distance, while considering vehicle capacity and battery life constraints.

The objectives of this research are as follows:

I. To propose Ant Colony Optimization (ACO) algorithm to solve the capacitated electric vehicle routing problem.
II. To apply the Taguchi method to examine the influence of key parameter variations on routing outcomes and the overall efficiency of the system

1.7 Scope of the Study

The study revolves on creating an improved system through the Ant Colony Optimization (ACO) algorithm with the integration of the Taguchi approach in locating the optimum routes and paths. This study has generated all the needed data, such as customer demand and distances between nodes by geography imaginations and simulation. These generated datasets make the study feasible to simulate realistic logistics, the location of depot, customers and charging station, the capacity of vehicles, and limited batteries. The models essentially emulate real life scenarios in logistic networks to ensure that the routing strategies for CEVs are fine-tuned regularly.

1.8 Significance of the Study

The importance of this study can be viewed from its ability to assist logistics firms in determining the routing of capacitated electric vehicles (CEVs). Firstly, the study intends to enhance the routes to be taken by vehicles in the hope of attaining better organization and execution of logistic services hence resulting to less energy expenditure in travelling, and increased reliability of the delivery services. This optimization can lead to considerable cost reductions for the logistics firm as it is directly associated with factors such as fuel consumption, maintenance and virtually almost all aspects of its operating expenditure. Such financial savings can be reinvested into other essential issues like enhancing fleet capabilities, growth in technology and customer relations.

Improving CEV routes also helped in improving on the environmental sustainability factor. The use of electric vehicles as well as the reduction of routes that are followed by transport fleets is another way that can help the logistics industry to minimize on the impact that it has on the environment. Such approach also helps to contribute to the international goals towards sustainability and create the awareness among the industry counterparts regarding the environmentally conscious ways of transportation. In addition, this research may have implications on other transport planning and management areas. Since the optimization methods discussed here apply to many fields, it only goes to show that this study can indeed help in improving the transportation networks of several industries.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

This literature review covers the main research on the CEVRP using the techniques adopted in planning the best routes for electric vehicles in logistics. It responds to the issues of restricted battery capacity and the amount of load that can be transported and studies the outcomes via exact methods, heuristics, and metaheuristics. It also recalls newer trends in hybrid methods and machine learning and their performance in the analysis. This analysis provides a comprehensive overview of the current approaches in the field, forming the basis for further research.

2.2 Capacitated Electric Vehicle Routing Problem in Logistic Industry

CEVRP is one of the important tasks in the logistics field as the key focuses are made on routes that may be better suited for EVs, and several factors, including the capacity of the cargoes and batteries, should be considered. This problem has emerged prominent due to the growth of EV freight industry and the drive for a green logistics. Several works have been presented to solve CEVRP, which use different techniques and goals in order to solve the problem. For example, the bi-objective mathematical model intends to minimize both the costs of routes and the delays, meanwhile implementing a meta-heuristic hybrid NSGA-II and

TLBO algorithm, thus showing high applicability regarding spacing and the rate of achievement to two objectives simultaneously (Zahedi et al., 2023). Another one is the extended ant colony optimization (EACO) in which pheromone matrix stores routing information, while the algorithm contains both coarse and fine construction methods addressing the issues with cargo and battery efficiently (Lin et al., 2022).

Moreover, combining DE algorithm with adaptive k-means and fuzzy approach it has been suggested for the improvement of initial global search and decision making and thereby improving the CEVRP solutions (Phu-Ang, 2023). Furthermore, a cluster-first, route-second heuristic approach relies on capacitated k-median and hierarchical constrained minimum weight matching clustering algorithms to take into account the EV range capabilities and enhance parcel deliveries; however, a Monte-Carlo Tree Search and Tabu Search handle route optimization (Carmen Barletta & Fallah, 2023). Thus, these various approaches underscore the necessity of CEVRP in optimizing 'green' and productivity efficiency of logistics' solutions for transport constraint and restricted battery range, whilst meeting time-sensitive demands.

2.2.1 Advantages and Challenges of Capacitated Electric Vehicle Routing Problem

When it comes to examining the CEVRP, there are distinct benefits and obstacles that are beneficial to the logistics sector. The first one is environmental since the use of electric cars help in reducing the emission of green-house gases cutting across the globe in the reduction of effects of climate change and meeting high set environmental standards (Mir Ehsan Hesam Sadati & Çatay, 2022; Zahedi et al., 2023). Also, the integration of EVs can result in a reduced operating expenditure primarily because the cost of both fuel and maintenance is lower than those of non-EVs in the long run (Zahedi et al., 2023). There is a demonstration of applying algorithms of the improved nature, for instance, EACO and hybrid VNS with Tabu Search as a means for solving goals of CEVRP which establishes the optimum routes and lessens the general use of distance as a way of boosting operation efficiency (Lin et al., 2022; Mir Ehsan Hesam Sadati & Çatay, 2022). Also, options like permitting the customers to use different addresses for a different time slot for delivery can enhance the customer satisfaction and the service delivery quality as well (Mir Ehsan Hesam Sadati & Çatay, 2022).

Nevertheless, the scheme also has several issues that are associated with the CEVRP. Some of the major challenges include; limited driving range, the long recharge times which could result in lengthy trips and complications in planning the route to coverage (Mavrovouniotis et al., 2020; Mavrovouniotis et al., 2022). Again, there is the variable energy consumption rate which is dependent on the weight of the cargo among other factors therefore competent algorithms have to be incorporated in constant calculation of the energy consumption rate (Mavrovouniotis et al., 2022). This fact complicates the construction of feasible solutions due to the interdependence between the required cargo capacity and the battery capacity, as evidenced by the development of the course and fine construction of the EACO algorithm (Lin et al., 2022).

Also, the problem is the integrate one since fleets can include both EVs and traditional vehicles, that makes it more challenging for the routing task to provide an effective solution, especially because of the dense clustering and routing heuristics necessary to reach the maximum number of deliveries with considering multiple cars' constraints (Carmen Barletta & Fallah, 2023). The application of these sophisticated approaches and techniques, including the Monte-Carlo Tree Search and Tabu Search, is known to mitigate these issues, yet their usage demands considerable computational resources and IT professionals' knowledge to launch and support (Carmen Barletta & Fallah, 2023; Mir Ehsan Hesam Sadati & Çatay, 2022).In addition, there is the concern of recharging structure since poor recharging facilities are often a major drawback that hinders effective functionality of the EVs in logistics (Mavrovouniotis et al., 2020). Nonetheless, the current continuing studies and developments on this matter involving the construction of new realistic and standard problem instances and the use of exact and metaheuristic approaches indicate that there is always a possibility of overcoming these obstacles and the enhancement and further adjustment of CEVRP solutions to the ever growing requirement of the logistics sector (Mavrovouniotis et al., 2020; Tahami et al., 2020). In conclusion, this paper has established that though the switch to capacitated electric vehicle routing in logistics has its challenges, it is worth it given the environmental and operational gains.

2.2.2 The Problems Faced in CEVRP

Even though the CEVRP brings such effective advantages to the logistics sector, some of the inefficiency need to be prepare always. There are several concerns that should be addressed, one of which is the issue of low electric vehicle battery, which inhibits long-distance driving and may require charging at arbitrary intervals. This is further compounded by the marked lack of charging stations for the batteries, thus hindering efficient distribution and logistics (Fazeli et al., 2024). However, having limited battery capacity and charging during the day also adds other challenges; the charging time means vehicle is stationary and the need to plan for the charging intervals. The changes in range caused by temperature fluctuations and the vehicle itself, including its load, velocity, and the quality of the roads also lead to inefficiency because they have a direct impact on the energy requirements of the vehicle and, as a rule, require recharging more often (Rastani & Çatay, 2023). Additionally, it is pertinent to point out that the load-carrying capacity of EVs depends on battery capacity, and failure to account for the load weight may have adverse repercussions often resulting in significant alterations in route and fleet planning and consequently leading to disruptions of services and overall higher costs (Rastani & Çatay, 2023). The selection of distribution channels is critical to costs or return on investment to the supply chain The present distribution channels fail to offer a comprehensive service package, thereby resulting in high costs of distribution and low productivity (Luan, 2024). There are problems with basic genetic algorithms in terms of stagnation and poor solutions, for which improved genetic algorithm and ant colony optimization algorithms have been introduced in order to optimize path pheromone concentration and sub functions for better distribution and reduction in cost (Lin et al., 2022; Luan, 2024).

However, to implement these algorithms in actual operations as proposed in this research, there is need for efficient data management systems as well as to intelligently schedule the flow of logistics vehicles to enhance distribution equity (Luan, 2024). Additionally, the requirement of having rescheduling procedures to accommodate precise points of downed service stations also complicate the CEVRP, which requires robust and dynamic planning approaches (X. Li et al., 2024). Applications of deep reinforcement learning algorithms have demonstrated the ability to minimize energy usage while at the same time have the potential for feasible routing; however, strictly algorithmic methods of this nature often demand enormous computational resources and up-to-date information to work efficiently (Tang et al., 2022). Speaking systematically, the inefficiencies in CEVRP are technical, geographical, where the conduction modes of EVs, the planning of reasonable routes and proper recharging stations, and the practical application of effective optimization and data management systems are problems related to CEVRP and the future sustainable development of logistics.

2.3 Algorithm of the Capacitated Electric Vehicle Routing Problem (CEVRP)

The Capacitated Electric Vehicle Routing Problem (CEVRP) is a realistic and challenging problem in the context of the transportation and logistics industry as the result of an increasing usage of electric vehicles due to their benefits in the aspects of the environment and costs (Asghari & Al-e-hashem, 2021). Therefore, the CEVRP poses source constraints different from conventional vehicle routing problems restricting the battery capacity, the recharging protocols, and the finite carrying capacity of the EVs. These aspects call for the integrated look at the route optimization strategy that considers the benefit of the company, costs, and impact on the environment (Fazeli et al., 2024). The following taxonomy explores the characteristics of the CEVRP such as problem, type of problem, objective functions, constraints, solution methodologies, implementing considerations, and application instances in detail to analyse and solve this complex problem in the logistics environment.

2.3.1 Problem Variants

2.3.1.1 Basic CEVRP

This is the basic formulation of the problem in which the objective is to identify the optimal routes for an electric vehicle (EVs) fleet. Every EV has its limited battery capacity and carrying capacity; it must determine the routes for serving the different demands of the customers.

2.3.1.2 CEVRP with Time Windows

In this variant of the problem setting, each customer is assigned a particular time window during which serving, in terms of deliveries or pickups, must take place. This complicates it further from the routing perspective because while selecting the routing the necessary time constraints cannot be violated while also at the same time adhering to the battery and cargo constraints.

2.3.1.3 CEVRP with Multiple Depots

In this problem there several depots, which indicate that vehicles can start and end in various places. This means that another planning is needed to determine where in the depots each of the vehicles should originate from and which depot they should return to in the most efficient manner as to conserve resources.

2.3.1.4 CEVRP with Stochastic Elements

This version helps to include different unpredictable aspects of the system like variable travel time, variability in the demand of the clients, and unpredictable availability of charging stations. It indicates that such uncertainties have to be met by solutions that are strong and dependable.

2.3.1.5 Dynamic CEVRP

For dynamic cases, some of the changes like new delivery requests, traffic conditions of the roads, or changes in the battery levels of the vehicles must be incorporated. The routing plan must be flexible enough, to be able to answer such kind of changes during the actual routing process.

2.3.2 Objective Functions

2.3.2.1 Minimization of Distance/Energy Consumption

The first objective is towards minimizing the total distance of the EVs or the total energy used. The combined goal is to find routes that minimize the total distance, considering that the constraint of each car's capacity and battery must not be exceeded. This problem has been approached by different methods as elucidated by the following researchers. For example, the MILP model can be used to formulate the routing problem in which various charging options for EVs and mixed fleets of vehicles can be utilized, and it was proved that vehicles of EV type will be used instead of CVs in last mile problems (Amiri, n.d.).

Other metaheuristic methods include the Hybrid Two-stage Heuristic Algorithm and Hybrid Variable Neighbourhood Search, which have been applied in tackling Green Vehicle Routing Problem to minimize the overall distance and distribution costs while incorporating instances of service station breakdowns (X. Li et al., 2024). In addition, clustering heuristics and adaptive algorithms, for example, the Adaptive Simulated Annealing algorithm, have been applied to achieve the increased speed and quality of the solution and to prevent disruptions to the routing plan (Pustilnik & Borrelli, 2024). This also caters the imprecise nature of customer demands through the utilization of fuzzy and intuitionistic fuzzy random variables influencing the selection minimizes the operational cost and maximizes the profit satisfying the gap between theoretical model and actual scenario (Singh et al., 2023).

As for the versions of Local Search based metaheuristic methods such as Multistart Local Search and Greedy Randomized Adaptive Search Procedure, the authors noted that both strategies have been reported to perform significantly well on real-life instances, owing to their systematic approach toward exploring combinatorial formulations of the problem identified (Matijević et al., 2024). Last of all, the differential evolutionary algorithm integrated with adaptive k-means and fuzzy methods has been claimed in the literature to yield superior performance as the initial solution search is enhanced, and the exploitation as well as exploration abilities are empowered (Phu-Ang, 2023). Taken together, these approaches underscore the need for the incorporation of enhanced decision-making approaches and well-defined models to enhance the overall success of minimization of total distance travelled, without compromising the requirement of satisfying customers' demands and constraint of vehicle capacity and battery in CEVRP.

2.3.2.2 Minimization of Operational Costs

Concentrates on minimizing the total expenses incurred in conducting the delivery operations. This can be things like fuel/electricity as a source of power, wear and tear on automobiles, wages paid, and anything that may be required in charging. The reduction of operational costs in capacitated electric vehicle routing problem, commonly referred to as E-VRP is complex due to several mainly constraints and factors as discussed below. This one is based on designing complex models of decision making that can be applied to the characteristics of electric vehicle demand, such as, for instance, comparatively short ranges and the necessity to recharge. For instance, the Hybrid Two-stage Heuristic Algorithm has been introduced towards minimizing the total distance of a planning route with consideration to distribution costs and service station mishaps and rescheduling techniques (X. Li et al., 2024). Another one is the realization of a bi-objective mathematical model that will have the goals: the minimization of the route costs, minimization of the delay of the vehicle arrival time with the help of a metaheuristic hybrid approach that included non-dominated sorting genetic algorithm and teaching-learning-based optimization (Zahedi et al., 2023). Moreover, the inclusion of driver wellness into the routing problem increases efficiency because of the decrease in failed delivery due to driver fatigue; an example by a hybrid Ant Colony System algorithm that optimizes timing of charging and rest (Su & Chen, 2023).

Furthermore, another mathematical continuous optimization technique like particle swarm optimization (PSO) and simulated annealing (SA) have been used to solve the capacitated VRP with an intention of minimizing the total distance travelled with good results, but for this, the PSO and SA, the solution has to be the converted into discrete one. Capacity constraints of charging stations also need to be addressed since failure to do so produces unrealistic solutions. Formulating the model in continuous time and the algorithm based on the generation of routes and their assembly in consecutive cycles proved useful for solving E-VRPs with nonlinear charging functions and at stations with limited capacities (Froger et al., 2022). In addition to that, the incorporation of new Information and Communications Technologies and optimization techniques, which include the Guided Local Search algorithm, will go a long way in enhancing the management of transportation networks and delivery issues, and will make sure that the capacity of a truck is never to be overloaded while at the same time making sure costs are sliced down to the minimum

(Mamoun et al., 2022). Some of the real-word applications like, Football Game Algorithm in a bottled drinking water company have confirmed the cost benefits and therefore the efficacy of these advanced algorithms in cutting down the costs (Alif et al., 2022).

Last of all, the improvement of environmental conditions due to the utilization of electric fleets cannot be refuted. Therefore, by making the necessary amendments to the routes of the vehicles: charge-discharge cycles, charging time, and the location of the charging stations, the companies realize a cut on costs, as well as a positive impact on the environment through the decrease in pollution and greenhouse emissions, enhancing the sustainable development . Thus, for capacitated electric vehicle routing problem where the aim primarily focuses on the minimization of operational costs, it was recommended that there is need to incorporate advanced algorithms to ensure effective solution.

2.3.2.3 Maximization of Service Level

Making sure that delivery of services occurs within the specified level of service delivery. Sometimes customer satisfaction is used as a common measure in logistics since it gives a clear picture of the delivery performance. Thus, the service level maximization in the capacitated electric vehicle routing problem (CEVRP) is solved with several constraints, such as those concerning vehicle capacity, driving range, and charging station accessibility. There are different approaches to work through complex problems and to address these issues in the best efficient way. For example, a new Hybrid Two-stage Heuristic Algorithm was used to minimize the total distance and distribution cost; also, to consider the Risk of failure of service stations and the consequent rescheduling of tasks (X. Li et al., 2024). Other strategy focuses on the bi-objective mathematical model on cost of route and total delay where the mixed meta-heuristic of NSGA-II and TLBO is used (Zahedi et al., 2023). Also, a heuristic approach to optimize the delivery of many parcels can prepare clusters of drop-offs according to the locations and weights and then prepare routes using a Monte-Carlo Tree Search and a Tabu Search (Carmen Barletta & Fallah, 2023). In the context of CEVRP the bilevel ant colony optimization algorithm breaks it into capacitated VRP and fixed route vehicle charging problems both of which are solved in an efficient manner to make the solution quality better through this effective combination of components (Jia et al., 2022a).

Besides, based on the charging station capacity, a continuous-time model formulation and an algorithmic framework that alternates between route construction and solution generation have been proved to be efficient to deal with E-VRPs characterized with nonlinear charging functions as well as capacitated stations (Froger et al., 2022). The differential evolutionary algorithm also used along with an adaptive k-mean algorithm and fuzzy techniques to search for the best solutions of the routing and to increase the efficiency of the DE by using the adaptive as well as decision making procedures (Phu-Ang, 2023). In addition, there is extended ant colony optimization (EACO) algorithm that utilizes pheromone matrix for storing routing data and the coarse and fine construction of routes for developing customer service sequences with recharging stations, which are demonstrated to comprise higher exploration ability and improved performance for exact routing construction (Lin et al., 2022). If mid-route recharging is not potentially required, time-constrained

CEVRPs can be solved with a mixed-integer optimization model, containing vehicle capacity, maximum allowed travel time and service time constraints, for e-commerce parcel delivery, and the indication that average vehicle miles travelled is within the BEV range, minimal BEV range impact studies have been reported (Cokyasar et al., 2023). Altogether, such diverse methodologies and algorithms enhance the service level by solving the various faced constraints as well as by improving several facets of the problem in CEVRPs.

2.3.3 Constraints

2.3.3.1 Battery Capacity Constraints

The problem of range arises from the battery storage in the EVs. Another objective of the routing plan is to guarantee that the vehicles can successfully undertake their entire routes without having any instances whereby the battery power would not be enough and would subsequently require recharge. The battery constraints are especially essential because they determine how far an EV can drive before recharging, so the distance must be factored into the Advanced RV routing problem. It has been suggested about these constraints, some methods that have been suggested to tackle some of them are; For instance, an extended ant colony optimization (EACO) algorithm utilizes pheromone matrix to store routing information and apply both coarse and fine route construct mechanism to construct customer service sequences which have the recharging stations; while in the fine routing mechanism, the battery capacity is adopted to evaluate the acuteness of vehicles' battery power (Lin et al., 2022). Another research also presents a variant of the E-VRP called E-VRP-NL-C that considers the number of charger at charging facilities and replaces the discrete time formulation with a continuous time one to capture better the time required at charging facilities which is used with Iterated Local Search and Branch-and-Cut to manage capacity constraints since the established model incorporates charging function that is nonlinear (Froger et al., 2022).

More research done on e-commerce parcel delivery truck routing also shows that mid-route recharging is not always possible due to capacities and maximum delivery time of the vehicle, showing in a study carried out that VMT per vehicle is approximately 72 miles and therefore; range of battery number does not affect performance index (Cokyasar et al., 2023). Other research activities include the application of mixed integer optimization formulation in solving time constrained capacitated VRPs especially for e-commerce parcel delivery vehicles; the analysis of the effects of maximum allowed travel time, service time, vehicle capacity, and BEV range on System-level performance indicators (Cokyasar et al., 2023). In addition, the vehicle routing problem of electric fleets entails battery capacity, charging time and place of charging to reduce the cost and time of distribution activities. Other works also use a clustering technique, DBSCAN, for generating feasible routes while other works such as Christofide's approximation algorithm also consider other constraints like capacitated vehicles and the distance (Choudhari et al., n.d.).

2.3.3.2 Vehicle Capacity Constraints

Every make of EV has an ultimate load-carrying capacity, expressed by either volume or mass. Distribution networks should be designed in such a way as to make sure that the total thrash does not come close to this capacity. The CEVRP is a potentially a major difficulty in logistics since many problems are bounded by twofold constraints of a vehicle load capacity and battery storage capacity. These factors dictate the need for appropriate strategies, which would enhance the rate of routing and delivery. The Hybrid Two-stage Heuristic Algorithm minimizing distribution costs and considering the short driving ranges and refuelling structures for alternative fuel-powered vehicles (X. Li et al., 2024). Another of the methods involves the integration of the differential evolution algorithm with adaptive K-means and fuzzy that improves the search of the best solutions concerning the restricted source of electric energy and goods' capacity of each vehicle (Phu-Ang, 2023). Ant colony optimization also presents the extended ant colony optimization (EACO) algorithm that too, having pheromone matrix for maintaining the routing information, and it also provides both coarse and fine route constructions for efficient management of cargoes and batteries accordingly (Lin et al., 2022).

Moreover, one more variant of the ant colony optimization algorithm considers CEVRP as a bilevel optimization problem, in which the capacitated VRP and the problem of fixed route vehicle charging are solved as two distinct problems to provide a more effective solving of the electricity constraints and the customer needs (Jia et al., 2022a). Hence, the cluster-first, route-second heuristic approach comprising the capacitated k-median and hierarchical constrained minimum weight matching clustering algorithms expands the solution space by considering range and capacity of EV's to enhance parcel delivery efficiency (Carmen Barletta & Fallah, 2023). Furthermore, the capacitated and time-constrained EVRP is solved by a two-stage solution construction method integrated into several types of the ACO algorithms regarding the practical problems of limited charging time and customers' urgent needs (Nie et al., 2022). Finally, the proposed cluster-first and route-second algorithm with integer linear programming for cluster identification and the DBSCAN algorithm for clustering yields feasible routes for different constraints such as the vehicles' capacity and distance, which can be used in real-life delivery networks (Choudhari et al., n.d.). Altogether, the proposed approaches stress out the need to account for the two objectives when solving CEVRP problems and showcase several types of methods for accomplishing routing under these considerations, further improving the effectiveness and sustainability of the EV logistics.

2.3.3.3 Time Window Constraints

Collection or drop offs must be done at certain times or time slots. Delaying or advancing the arrival or service time within these time windows may be costly through fines or reduced customer gratification. The foremost goal is to achieve low operational cost especially for fuel and penalties, with respect to time windows of deliveries and pickups (Benotmane et al., 2024). To this end, the bi-objective mathematical model has been presented costing the routes, while avoiding delays of vehicles arrival in depots by minimizing the route costs and delay of vehicle arrival at depots; With a sincere metaheuristic algorithm, a promising result was obtained as the spacing and rate of achievement to objectives simultaneously (Zahedi et al., 2023). Other techniques, like the Actor-Critic architecture also used in reinforcement learning have also been used in optimizing routing efficiency since the methods able to make optimal sequences that fulfil the time window constraints of the environment (Wang et al., 2023).

Furthermore, A two-dimensional loading and time windows employs a multi-stage algorithm to consolidate all the customers which those Vehicles visiting into fewer ones and applying an enhanced ant colony algorithm in the CVRPTW route optimization process (Zhou et al., 2022). The extension of charging functions and charging stations as capacitated adds another layer of difficulty to the E-VRP and requires the use of a continuous time model as well as an iterative algorithm to ensure that the solutions generated are of high quality (Froger et al., 2022). In e-commerce parcel delivery, capacitated time-dependent vehicle routing problem has been solved using a mixed-integer optimization model and identified that vehicle capacity and service time are the most influential factors of OR while having negligible sensitivity to BEV range as average VMT falls in this range (Cokyasar et al., 2023). Pay attention to the driver's wellbeing when devising the E-VRP and regularly, an enhanced Ant Colony System algorithm has been created. It combines routing, charging, and rest schedules to satisfy both the vehicles and the driver's fatigue constraints (Su & Chen, 2023).

In addition, the integration of reinforcement learning, policy rollouts, and satisfiability solvers has also proved to be significant in generating high-quality solutions with less time compared to those of metaheuristic algorithms (Khadilkar, 2022). The application of k-means clustering and Tabu search in dynamic capacitated routing problems with time windows has also revealed the decrease in the economic and energy costs, noting the significance of proper customer clustering and the route search (Benotmane et al., 2024). In addition, mixed-integer linear programming models (MILP) and general heuristics has been used to minimise one's route time and bound route times within a time window, which infers the NP-Hard nature of the problem of utilising heuristics to solve it efficiently (Zhou et al., 2022). Altogether, these works highlight time windows' limitations in capacitated electric vehicle routing further, asserting a synergistic integration of state-of-art algorithms and effective time-constrained heuristics for enhancing EV logistics' sustainable solutions.

2.3.3.4 Charging Constraints

The charging of such vehicles is also a consideration when planning for a given route in relation to availability of such services, its location and time taken to recharge the vehicles. The charging infrastructure should be optimally used to ensure that dwell times are kept to a minimum. An important challenge when solving CEVRP is the limited number of charging stations and rather short cruising range of electric vehicles that requires a proper planning of service orders and recharging timetables (Jia et al., 2022a). The more common applied methods of planning the route for an EV can have high computational complexity or lower efficiency when applied to large-scale problems. To solve these problems, some approaches have been developed. For instance, a Deep Reinforcement Learning-based methodology can be established to find near-optimality by identifying a sequence of actions and acquiring a feasible route without the requirement of re-training for each new problem set. This method also involves a layer of generating a charging scheme on the path, and this is regardless of the topology of road network and the types of EV (Y. Zhang et al., 2023). Another approach indicates an intermediate quantum strategy between the classical and quantum methods based on the use of multi-level carriers of quantum information in sample-based search to minimize the search area and the number of iterations when using quantum mechanisms. Promising outcomes have been revealed by this method in the toy instances of the EV charging and routing problem (De Andoin et al., 2023).

Also, the issue of integrating EVs with public transportation schedules has been solved, considering the limited ability of charging spots and partial charging. This approach has however been used to obtain integer feasible solutions by the path based binary program and the use of column generation techniques, pricing strategies such as ‘price-and-branch’ and ‘diving heuristic’ have been effective in solving large instances in reasonable time (de Vos et al., 2024). Besides, the factors affecting the cost of EV charging include charger cost, energy cost, and battery capacity as through solving the EV supply equipment location and allocation utilizing a mixed-integer linear program, clustering approach, and a genetic algorithm (Davatgari et al., 2024). The incorporation of drivers’ welfare into EVRP has been also investigated, where authors introduced a blend of the Ant Colony System algorithm to determine the optimal amount of charging time and rest pause to boost safety and productivity on the road (Su & Chen, 2023). Moreover, a bidirectional binary ant colony optimization algorithm has been proposed to solve the EVRP with capacitated and temporal constraints and proved efficiency in producing feasibility solutions (Nie et al., 2022).

2.3.3.5 Route Constraints

Some tracks have specific constraints like maxima load, closures of certain roads, or state of traffic. These must be taking into consideration, while planning the routes. A challenge that is characteristic of CEVRP is the capacity of EV batteries that results in a limited driving range of EVs, which, thus calls for accurate identification of break points in the routes. This is fixed by including recharging points on the vehicle routing plans, in which a vehicle will require partial or full recharge depending on the charge left on the battery, and the distance to the next customer or depot (Bezzi et al., 2023). Further, the cargo carrying capability of the EVs poses another constraint in which only a certain number of products can be transported in the vehicle and hence the optimization has to efficiently distribute the load between several vehicles (Chanachan et al., 2023). To address such limitations, different methods have been suggested. For example, a Mixed Integer Program (MIP) approach has been employed to solve while exactly reformulating the problem as a collection of simpler sub-problems so that re-planning could be efficient when the scenario changed (Pustilnik & Borrelli, 2024). The second one is used in a Hybrid Two-stage Heuristic Algorithm and is focused at minimizing total route distance, and at the same time, minimizing distribution cost while at the same time being able to adapt to service station failures (X. Li et al., 2024). Other optimization techniques that have been considered include the genetic algorithms and differential evolutionary algorithms where methods like adaptive K-means clustering and fuzzy decision making have also been employed with a motive of improving the search of the solutions (Chanachan et al., 2023).

Furthermore, deep reinforcement learning has helped in the design of a two-layer algorithm that identifies almost the best combinations of the actions and generates practical routes without the requirement of re-training for every problem instance (Y. Zhang et al., 2023). Extensions of the ACO algorithms involve pheromone matrices, which contains routing information of the customers and charging stations design of routes with respect to the capabilities of the cargo as well as battery (Lin et al., 2022). A bilevel optimization approach has also been proposed, dividing the problem into two sub-levels: One of the models concentrates on the capacitated VRP and the second model deals with the fixed route vehicle charging problem; thus enhances the overall solution by efficiently managing between the two parts (Jia et al., 2022a).

Last but not the least is the aspect introduced by time-dependent charging function indicating that long charging time is incompatible with the fast and efficient service provision. To this end, a two-stage solution construction procedure for ACO has been designed and integrated into many ACO algorithms for generating feasible solutions that meet the capacity constraints and charging time interval (Nie et al., 2022). Such methodologies demonstrate that the task of CEVRP is complex and multifaceted, and the approaches to its solution are being designed by practitioners.

2.4 Solution Approaches for CEVRP

2.4.1 Exact Algorithms

An exact algorithm is a very sound and distinct method developed mainly for solving optimization problems and guaranteeing the best solution. Applied with the help of optimisation methods, this strategy implies defining the problem's objective function and constraints initially. The best solution is then achieved when either the minimising or the maximising of the objective function is achieved while fulfilling all the constraints. In the context of the Capacitated Electric Vehicle Routing Problem (CEVRP), exact algorithms ensure to find the optimum routes to be traversed by the electric vehicles, as measured by distance, or cost, or any other desirable constraint given the vehicle capacity and battery capacity (Rand, 2024).

2.4.1.1. Branch and Bound (B&B)

Branch and Bound (B&B) is an algorithm in which systematic exploration of the solution space is carried out by dividing the problem into subproblems, that is branching and calculating the lower bound for each subproblem. It takes the whole problem and divides it into the sub problems, the sub problems referring to portions of the solution space. For each subproblem, the algorithm estimates just an upper as well as a lower bound of the optimal value. If the bound computed for a subproblem shows that it cannot give a better solution than the current best solution discovered, the sub-problem is pruned or discarded. This process continues until all the sub problems are solved or pruned and guarantee that the best solution is developed (Vu et al., 2007). B&B works but it takes relatively long time when dealing with large problem because of the number of subproblems that it has.

2.4.1.2. Branch and Cut (B&C)

The method B&C is a combination of branch and bound with cutting plane. Jim cut planes are linear inequalities that define addition to the problem in order to make the feasible region smaller so that it doesn’t contain any fractional solutions that would not be acceptable in the integer problem. The algorithm starts from a relax of the original problem and attacks it as a linear programming problem. If the solution given by the algorithm is not integer, then the cuts are generated and incorporated within the problem from where the fractional solution needs to be eliminated in order to attain a tighter relaxation. Thus, the described process is repeated with the switch between branching and adding cuts until reaching an integer optimal solution (Jiao et al., 2014). Due to this reason B&C is more efficient than the pure B&B for a number of problems because the cuts assist in the reduction of the solution space.

2.4.1.3. Dynamic Programming (DP)

DP divides the CEVRP into subproblems and solves them in a recursive manner due to the nature of the problem. It employs the optimality fundamental that asserts that an optimal solution of the main problem affords optimal solutions to the sub-problems. DP usually entails partitioning a given problem into subproblems in a state space, in which each state is associated with a subproblem and a recursive formula (commonly, the Bellman equation) that defines the solution of a state out of the subsequent states' solutions (Treiber, 2013). Thus, by storing the solutions of subproblems (memorization), DP is able to solve large and complicated problems in as efficient a manner as possible. However, the approach can be also weak with large size problems since the state space increases exponentially when the problem size increases, it can be called curse of dimensionality.

2.4.2. Heuristic Algorithms

Heuristic algorithms are the general approach to solving a problem wherein one comes up with good if not the best solutions in given time. These methods are especially relevant for the problems like Capacitated Electric Vehicle Routing Problem (CEVRP), when the exact solution could be considered as computationally unfeasible. Heuristics supply workable solutions by making the problem easier or using the cut and dried formulas. These methods do not guarantee that the best solution for any given problem is found but are ideal for cases where a satisfactory solution is wanted in a short time and for large problems (Barak, 2013).

2.4.2.1 Nearest Neighbour (NN)

Nearest Neighbour (NN) heuristic is a greedy algorithm in which the route is constructed by repeatedly adding customer nearest to the current one not included in the route yet. This method is also easy and fast to implement but its efficiency is a little poor due to the lack of the global view of the problem (Mohammed et al., 2017).

Step 1 : Initialization: Perhaps begin with the vehicle devoid of any passenger from the depot.

Step 2 : Selection: Starting from the current position, identify the closest customer that has not been served and is within reach based on the vehicle's capacity and battery power.

Step 3 : Visit: Go to the chosen customer, perform the delivery of goods, decrease the quantity as well as the battery power in the vehicle.

Step 4 : Repeat: Keep on with the selection and visit steps until all customers have been served, or the vehicle must get back to the depot to replenish it power source or supplies.

Step 5 : Return to Depot: Finally, if required, trace back to the depot to cover the customers which were not served earlier for that route and then possibly start a new cycle for a new shorter route for remaining number of customers if any.

Even though its implementation is quite basic, NN may result to suboptimal paths, as the closest neighbour is selected based on current position without considering the subsequent decisions.

2.4.2.2 Savings Algorithm

The Savings Algorithm is a better heuristic proposed by Clarke and Wright Starts with a route of any customer and keeps on grouping the routes to minimize the total distance.

Step 1 : Initialization: Begin with each customer in its own route, each route can be defined as a trip from the depot to the customer and back to the depot.

Step 2 : Compute Savings: For each pair of customers i and j, calculate the savings from combining their routes into a single route. The savings S_{ij} is given by:

$$S_{ij} = d_{depot,i} + d_{j,depot} - d_{i,j} \tag{2.1}$$

Where:

- $d_{depot,i}$ is the distance from the depot to customer i
- $d_{j,depot}$ is the distance from customer j to the depot
- $d_{i,j}$ is the distance between customers i and j

Step 3 : Sort Savings: Rank the savings in descending order, meaning, arrange them in an order in which the first figure is greater than the next figure.

Step 4 : Merge Routes: Greedily combine the best two routes which results in the updated route that violates the limitation out of the capacity and battery's power.

Step 5 : Update Routes: Keep on superimposing until no more profitable overlaps are observable.

The Savings Algorithm takes slightly more time to execute than the Nearest Neighbour heuristic but provides better results because it takes into consideration the general effects of route amalgamation.

In all, it can be said that heuristic algorithms like Nearest Neighbour and the Savings Algorithm are useful in developing solutions to the CEVRP. However, while NN is simple and efficient, the Savings Algorithm usually provides slightly improved solutions utilizing the possible savings from consolidation of the routes.

2.4.3 Metaheuristic Algorithms

Metaheuristic is another class of algorithms that are independent of the problem being solved and are also at a higher level that directs other heuristics in order to exhaust the solution space. Clauses of differentiation are particularly advantageous in numerous optimization issues such as the Capacitated Electric Vehicle Routing Problem (CEVRP), where the search space’s vastness makes it difficult to use conventional heuristics. Metaheuristics are used to obtain near-optimal solutions in reasonable amount of time by trading exploration of new solution regions and exploitation of the known promising regions (S. Zhang et al., 2018).

2.4.3.1. Genetic Algorithms

Genetic Algorithms are based on the natural selection and genetics processes. Developing a population of possible solutions through generations in a process commonly known as evolution (Shukla et al., 2015).

Step 1 : Initialization: Create the initial population randomly (this means the first set of solutions or routes randomly).

Step 2 : Selection: Assess the feasibilities of the solutions in relation to the objective function, meaning the total amount of travelled distance for example. Choose a portion of the population with better solutions with reference to a set of fitness criteria.

Step 3 : Crossover: Blend the pairs of selected solutions (parents) to create new one (children) sharing segments of route of the parents' solutions. This resembles how crossover occurs in the process of genetics in a population of organisms.

Step 4 : Mutation: It is beneficial to add some random changes to some solutions in order to avoid the trapping in local optima as well as to increase the variety of the population.

Step 5 : Replacement: Introduce the new generation replacing the old population with them, taking with them the best solutions which can be taken forward.

Step 6 : Iteration: Perform the selection, crossover and mutation, and replacement steps again and again for the required number of generations or until the algorithm comes to its convergence.

2.4.3.2 Simulated Annealing

Simulated Annealing is based on the annealing process of metallurgy, in which an object is heated to a high temperature and then gradually cooled to eliminate cracks (Triki et al., 2005).

Step 1 : Initialization: It is assumed the first solution, and first temperature are given.

Step 2 : Neighbour Generation: Propose a neighbour solution by modifying the current solution by exchanging two customers in a route.

Step 3 : Acceptance Criterion: Computes, whether to move to the neighbour solution based on its quality and some probability which is related to the temperature. Actually, worse solutions may be considered as legitimate to avoid local optima, while the probability decreases with lowering the temperature.

Step 4 : Cooling Schedule: Stepwise decrease the temperature as defined in the timetable.

Step 5 : Iteration: Step: Repeat the neighbour generation and acceptance until the system cools down to a value that is less than the stopping criterion.

SA is good at escaping local optima but it’s necessary the proper tuning of the cooling schedule.

2.4.3.3 Ant Colony Optimization

Ant Colony Optimization (ACO) is based on the food search strategy of ants, which select the shortest routes which aromatize by putting pheromones (Mukherjee & Acharyya, 2011).

Step 1 : Initialization: Set the starting value of pheromone on all arcs to some small positive constant value.

Step 2 : Ant Simulation: Use several fake ants to build solutions by transferring from node to node based on pheromone strength and other information (for instance length).

Step 3 : Pheromone Update: Revise the pheromone amount regarding the quality of the solutions that have been obtained. Good solutions get more pheromone sprayed on them, emphasizing those routes.

Step 4 : Evaporation: Use pheromone evaporation to decrease the impact of previous paths and introduce other paths of navigation.

Step 5 : Iteration: Perform the ant simulation and pheromone update until certain number of iterations or until the problem converges.

ACO also performs exceptionally well for the routing problems since it also refers to the good paths with the help of positive feedback.

2.4.3.4 Particle Swarm Optimization

Particle Swarm Optimization (PSO) belongs to the family of swarm intelligence, like the process of birds' flocking or fish schooling. It just finds a solution to a problem by using a population of searching objects (particles) roaming in the solutions domain (Guo et al., 2008).

Initialization : Start a population of particles anywhere in the solution area randomly with arbitrarily assigned velocities.

Evaluation : Check whether each particle is fit to the current problem as per the objective function.

Update Velocities : Extract each particle current position, current best-known position of each particle and the global best position of the swarm of particles and then update each particle velocities based on its best-known position, its neighbour best and the best global positions obtained by the swarm of particles.

Update Positions : Advance each particle to the next position based on its velocities after calculation of angle and magnitude.

Iteration : Continue repeating the evaluation of the velocity, the update of the velocity and the update of the position steps until the algorithm halts on a proposed condition.

Thus, PSO is best used for continuous optimization problems having been adopted to fit discrete problems such as CEVRP.

All in all, metaheuristic algorithms including GA, SA, TS, ACO and PSO are promising to deal with the CEVRP by balancing the exploration and exploitation to obtain good solutions in a proper time. Both methods have certain distinctive features and advantages that do not allow to use them instead of each other but make them applicable for solving different optimization problems.

2.4.4 Hybrid Method

Hybrid method incorporate features and properties of different optimization methods, avoiding the weaknesses of certain algorithms. On one hand, as a number of algorithms in composite algorithm are usually utilized to work simultaneously or in combination with each other, hybrid methods look forward to realizing higher efficiency, higher solution quality, as well as higher robustness than a single method (Ting et al., 2015). In general, hybrid approaches can reach efficient solutions and high accuracy to face the Capacitated Electric Vehicle Routing Problem (CEVRP) using the exploration capability of metaheuristic and the precision of exact method.

2.4.4.1 Hybrid Metaheuristics

When two or more metaheuristic algorithms are combined with each other, then the resultant method is known as the hybrid metaheuristics which offers better scope for searching between exploration and exploitation. These combinations can increase and/or decrease the convergence rate and the quality of the solution (Shami et al., 2022). Common hybrid metaheuristic approaches include:

GA + SA : Genetic Algorithms (GA) produce a population of solutions through crossover and mutations; on the other hand, Simulated Annealing (SA) polishes up the solution from the Genetic Algorithm stage in such a manner that it does not get stuck at local optima. As for the application of GA, one can use it to evolve the population of solutions, while applying SA to the selected solution to improve it.

GA + ACO : GA can make a heterogeneous first generation and ACO can be applied to make each solution better by mimicking the behaviour of ants during search for food. It is believed that the pheromone trails that ants lay can lead the GA in choosing better parents for crossover.

TS + SA : TS can give a global survey of the solution space and eliminate cycles in the tabu list, and SA can make final modifications to the solutions by accepting worse solutions from time to time.

PSO + GA : PSO can show good performance in finding zones of the solution space, often effectively, while GA is able to maintain the population diverse and not to converge too early due to crossover and mutation.

2.4.4.2 Hybrid Exact-Metaheuristic Approaches

In hybrid exact-metaheuristic strategies, applicability of metaheuristics that provide global search is incorporated with the fineness of exact strategies. These approaches may enhance the derivation of high-quality solutions through metaheuristic to search for the possible solution and exact algorithms to fine tune the solutions (Muazu et al., 2022). Common hybrid exact-metaheuristic approaches include:

Metaheuristic for Initialization + Exact for Refinement : Jiang et al., (2010) used a metaheuristic algorithm that would help in creating the basic solution or a good solution for the problem. An exact algorithm (e. g., Branch and Bound, Branch and Cut) then comes in to tweak this solution to the optimum solution. For example, ACO can be used to generate a good initial solution, and MILP can be applied for improving the solution step by step starting from the point obtained.

Exact Algorithms with Metaheuristic Enhancements : Extend the application of metaheuristic templates on involving them into exact algorithms since they are usually faster. For instance, incorporating cutting planes generated by metaheuristic solutions into Branch and Cut format contributes to a better way of shrinking the feasible region. Likewise, the bounds created by metaheuristics can be incorporated in Branch and Bound to hasten the pruning procedure.

Decomposition Approaches : Deduce one component of the problem and find the solution to the component using a different approach from that of the other component. For instance, in the master problem, a metaheuristic can be applied while in the subproblems, exact methods can be used. This approach is frequently utilized in column generation methods: a metaheuristic might generate columns (routes), and an exact algorithm will solve the restricted master problem.

Metaheuristic for Large Neighbourhood Search : A metaheuristic is used to search a large number of solutions in the vicinity of the current solution and an exact algorithm fine tunes the best solutions within the local neighbourhoods.

For instance, a TS can create a neighbourhood and an exact algorithm can then calculate the best solution within that neighbourhood.

2.4.5 Decomposition Methods

Decomposition methods disperse a large and difficult optimization problem into several easier and probable sub-problems. From (Karabuk, 2009), these methods are especially helpful for such problems like Capacitated Electric Vehicle Routing Problem (CEVRP) for which the feasible region is enormous. The work in decomposition methods can be better and more efficient, as they solve subproblems in an iterative or parallel manner. Two methods which are in a common use for this purpose are Column Generation and Lagrangian Relaxation.

2.4.5.1. Column Generation

Column Generation is a method applied to deal with big linear programs if the number of variables is extremely high. It breaks down the problem into master problem and sub-problems, and in each iteration, it only generates the promising variables or 'columns' unlike in other approaches where all variables are considered and traversed through (Rodrigues & Yamashita, 2010).

Step 1 : Initialization: Begin with a restricted master problem (RMP) that has only a part of the variables (columns) incorporated in it. These columns for the initial profile can be derived based on heuristics or can be randomly assigned.

Steo 2 : Solve Master Problem: The RMP is to be solved in order to get the dual prices (the shadow prices of the constraints).

Step 3 : Solve Subproblem: Solving this subproblem using the information of dual prices derived from the RMP will reveal new columns (variables) that can optimally add to the solution. Regarding the CEVRP, the subproblem typically entails identifying the route that will be most economical in terms of the dual prices from the perspective of the vehicle under consideration.

Step 4 : Column Selection: If the subproblem determines some of the columns as having a negative reduced cost, which means they could potentially improve the objective value, incorporate them into the RMP.

Step 5 : Iterate: Continuously solve the RMP and the subproblems incorporating new columns in the RMP until no more negative reduced cost columns are identified.

Step 6 : Optimality: Since one cannot continue to look for better columns that can improve the current solution to the RMP, the current solution is the best for the original problem.

The use of Column Generation in vehicle routing problems is suitable because it can accommodate the numerous potential routes (variables), given by the number of distribution centers and vehicles as limited variables while attending to numerous potential variables.

2.4.5.2 Lagrangian Relaxation

Lagrangian Relaxation uses some constraints of the optimization problem, containing them into the objective function by using penalty terms known as Lagrangian multipliers. This of course alters the original problem into a simpler one that can easily be solved (Chen et al., 2010).

Step 1 : Relax Constraints: Determine which constraints should be eased (for example capacity or route constraints). Relocate these constrains into the objective function complemented with some associated penalty terms.

Step 2 : Formulate Lagrangian Problem: Sum up the penalty terms with the objective function to get the Lagrangian function. The first allows for getting a new objective function equal to the sum of the initial objective plus the sum of the relaxed constraints multiplied by their Lagrangian multipliers.

Step 3 : Solve Subproblem: Solve the resulting Lagrangian subproblem which usually is simpler than the original because it includes fewer constrains of simpler structure.

Step 4 : Update Multipliers: Update the Lagrangian multipliers through the solution of the subproblem to obtain better estimates of the constraint Lagrangian. This is always done by subgradient optimization or other iterative optimizers and so on.

Step 5 : Iterate: Solve the subproblem again and update all the multipliers as is done until the convergence of the solution. As a result, the solutions obtained are dual solutions that give the minimum value of the original problem.

Step 6 : Primal Solution: Build a realistic solution to the original problem based on the obtained dual solution and use the heuristic modifications if necessary. Thus, the quality of this solution gives an upper bound.

Step 7 : Gap Reduction: Employ the bounds to design more iterations or heuristic refinements to narrow down the difference between upper and lower bounds.

As for CEVRP, Lagrangian Relaxation is helpful because it allows to transform the complex constraints, make the problem easier to solve, and obtain helpful bounds and conclusions.

2.4.6 Mathematical Programming

Things such as mathematical programming can be described as sub discipline of optimization performing the function of formulating problems in the language of mathematics which implies that there is an objective function to be maximized or minimized and there are constraints to be placed on the outcome. Such constraints can depict certain factors or conditions in the real world like capacity, demand, or restriction to perform operations. Thus, the goal is to select an optimal solution from the solutions, which can be obtained only within the confines of this feasible region (Pujowidianto, 2017). Two classes of mathematical programming, which are used in solving Capacitated Electric Vehicle Routing Problem (CEVRP) are Mixed-Integer Linear Programming (MILP) and Constraint Programming.

2.4.6.1 Constraint Programming

Constraint Programming is a distinct paradigm which states the constraints and searches for solutions that are achievable with respect to all stated constraints. It is noteworthy that Constraint Programming is more effective for those problems with such properties as complex combinatorial structures and logical constraints (Kovacs et al., 2015).

Formulation:

Variables : Decisions can be quantified and represented as variables like the route that is to be taken by the vehicle or sequence at which the customers are to be visited.

Domains : Operationalize the values of the variables which it will contain by defining the possible values of the variables.

Constraints:

All-Different Constraints : Make sure you go to the each customer only once.

Cumulative Constraints : Control the loading capacity of the vehicles for a given route as well as the battery charge.

Precedence Constraints : Determine how the all routing and the operational requirements dictate the order in which the different customers are visited.

Logical Constraints : Avoid conditional dependencies and determine the subtle rules.

2.4.7 Machine Learning

The foundation of most Machine Learning methodologies is based on the use of data to tackle difficult optimization issues such as the Capacitated Electric Vehicle Routing Problem (CEVRP). As indicated, unlike the conventional algorithms that work through a set of rules or mathematical models, ML uses patterns obtained from previous data to make the right decisions or predictions (Valadarsky et al., 2017). These models are suitable for big data, capable to learn and readjust themselves after a certain time of their running and can give best solutions even if the problem has a high nonlinear nature. Reinforcement Learning and Neural Networks have been exemplified widely when it comes to routing problems in ML.

2.4.7.1 Reinforcement Learning

Reinforcement Learning is a branch or category of machine learning in which an agent has to decide an action in the environment. It is feedback based and the agent is embodied in a bank that gets a certain amount of reward or penalty depending on the actions it takes, and the overall aim is to maximize the net worth of the rewards over time (Y. Li, 2018).

Components:

Agent : The decision-maker in this case is for example in the context of vehicle routing, the algorithm itself.

Environment : The environment that surrounds the agent, that is, refers to the setting space that is consists of the road network, the customer locations, etc.

State : The situation of the agent in the given environment at the time the information was transmitted (e. g., the current position of the vehicle, the level of battery charge, or the load – carrying capacity).

Action : The resolution that has possibly been made by the agent (e, g, the next customer to arrive).

Reward : The information that one get after implementing an action, for instance, negative feedback upon using longer routes, and positive feedback upon using efficient routes.

Policy : The plan the actual agent uses to decide on its actions dependent on the present status (e. g, neural network, mapping status to action).

Process:

Exploration : In this mode the agent actuates to observe the stimuli it gets from the environment for possible change of results.

Exploitation : An agent uses the knowledge to get the most out of an environment according to the learned policy.

Learning : The agent then revises its current policy by the received rewards and employ methods like Q-learning DQN or Policy Gradient.

Iteration : Thus, with successive interactions of the agent with the environment, improving them using the policy.

2.4.7.2 Neural Networks

Neural Networks is a subfield of artificial intelligence that is based on the usage of multi-layer perceptron that incorporates many neurons. It is most effective in Pattern recognition, Regression and Classification problems. The clearly defined application of the CEVRP means that neural networks can be used in this context for the prediction of the best routes, the estimation of the time required to complete these routes, and in the operations of the decision-making body (Volna, 2016).

Components:

Input Layer : Accepts the unprocessed information in the problem (e. g. coordinates of customers, distance, capacity etc.).

Hidden Layers : Hidden layers in which the weights and the activation functions are used to transform the input data.

Output Layer : Returns the final output (e. g. , the expected route, the approximate price).

Types of Neural Networks:

Feedforward Neural Networks : Predictive basic form where nodes do not create any circuit; useful in simple prediction or classification.

Convolutional Neural Networks : Primarily used for data that has a spatial dimension – spatial analysts; important for those with visual or geographical information.

Recurrent Neural Networks : LSTM for dealing with input data in which previous data must be remembered; helpful when the data have a time-series or sequential decision making.

Graph Neural Networks : Originally intended for graph data; extensively used in routing problems where the nature of the road network is already a graph.

Training Process:

Data Collection : Collect records of the routes, travel time, customer needs/requirements, etc.

Preprocessing : Prepare it to be used with some algorithms: standardization and integration of compelling data for training.

Model Training : They include the backpropagation algorithm and the gradient descent algorithm that help to adjust the weights and reduce errors between estimates and actual results.

Validation and Testing : Cross-validate the model with the tests on different datasets in order to confirm good generalization to new data.

2.5. Selected Solution Approach For CEVRP: Ant Colony Optimization (ACO)

The capacitated electric vehicle routing problem (CEVRP) is a highly suitable case for ACO because this approach is characterized as flexible, adaptive and capable to effectively address the complex optimization tasks. ACO algorithms resemble the functioning of ants that have efficient way-finding techniques inherent to pheromone agents. This characteristic proves particularly useful within the context of CEVRP which considers equally numerous constraints such as, for instance, the vehicle capacities, battery capacities, and the available charging stations. Due to this flexibility, the ACO algorithm is a reliable solution to the problem identified by the constraints above . For example, there is an extended ACO (EACO) algorithm that stores the routing information of customers and charging stations in the pheromone matrix to construct solutions through the course and fine construction (Lin et al., 2022). Thus, the use of both weights to optimize the search space and utilization of carry capacity for cargo and battery strength improves the algorithm’s exploration and exploitation capabilities and the quality of the solutions found.

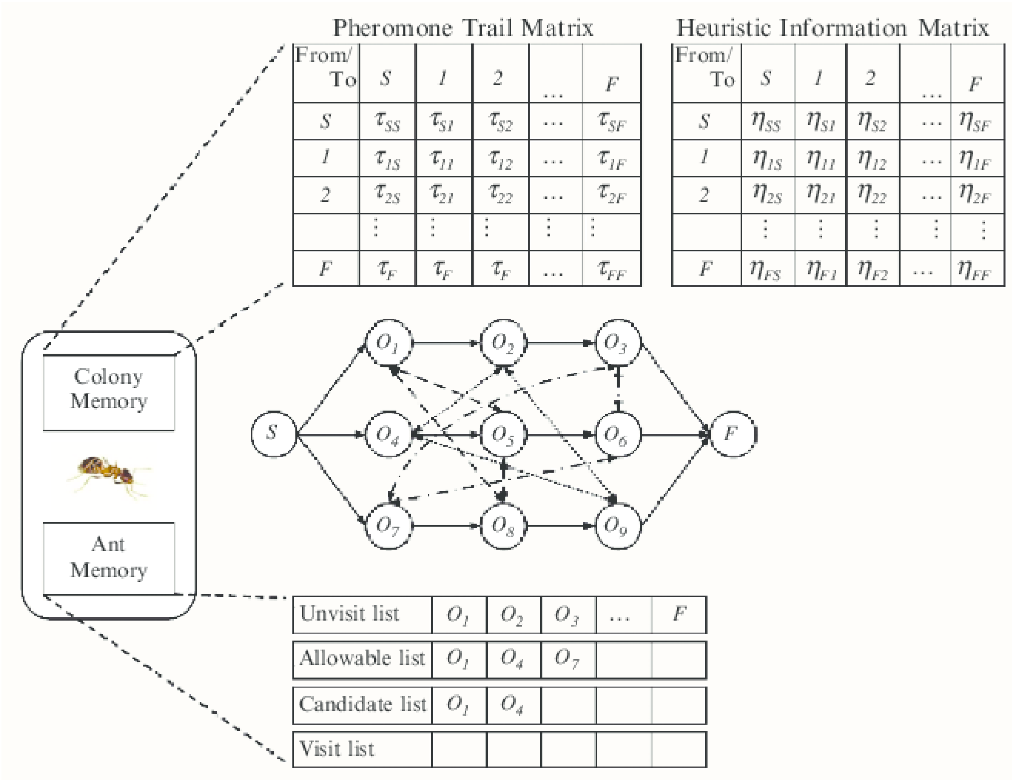

Figure 2.1 General Idea for Pheromone Matrix (Udomsakdigool & Kachitvichyanukul, 2008)

Additionally, exploration capability and effectively avoids the problem of falling into local optima to increase the accuracy of the solution (Zhao, 2023). Furthermore, the confidence-based bilevel ACO algorithm of the work simplifies the problem into upper and lower levels of problem solving through confidence thresholds that removes poor service sequences and fine tunes the recharging sequences to provide the best solutions that form benchmarks for updating previous best solutions (Jia et al., 2022b). The enhanced ant colony system (EACS) algorithm also proves its efficiency with the help of K-nearest neighbour (KNN) for primary solutions and k-opt algorithms for sub paths, that help in increasing the diversity and at the same time reduces the computational time (Ahmed et al., 2023).

Moreover, subsequent developments of ACO algorithms like the adaptations of the pheromone volatile factors and the customer selection process solves the problems of early convergence and local search as applied to big-scale problems (Ma & Liu, 2024). In this context, it is also crucial to mention that the ACO algorithm used in this work is the flexible one, which can work with different types of VRP variants, including backhaul datasets and multiple constraints, which makes this approach suitable for the CEVRP problem (Tablada et al., 2024). Finally, with the new insertion operations dealing between the nodes and paths, which greatly enhance the ACO algorithm and a faster convergence by selective rewarding of pheromones, the routes are optimized and the overall delivery costs is reduced to its minimum (Deng & Wu, 2023). Overall, the above-developed ACO algorithms establish their ability to effectively deal with the stakeholders' issues of CEVRP, thus becoming the optimal solution for this complicated global optimization problem.

2.5.1 Why ACO is Suitable for CEVRP

2.5.1.1 Adaptive Learning

When it comes to the adaptively learning in ant colony optimization (ACO) for capacitated electric vehicle routing problem (CEVRP), can be defined as the approach to modify and upgrade the algorithm's performance flexibility according to the new data which appears with time. The CEVRP is a diverse problem that must consider customer service along with vehicle recharging, and hence the conventional route cannot be accurate. As a result, many improvements to the ACO algorithm have been suggested by the researchers. For example, an enhanced ant colony optimization (E-ACO) algorithm introduces the traffic flow prediction and energy consumption model of each electric vehicle (EV) to redesign the heuristic factors and state transition rules to improve the accuracy and efficiency of path planning ("Dynamic Energy-Efficient Path Planning for Electric Vehicles Using an Enhanced Ant Colony Algorithm," 2024).

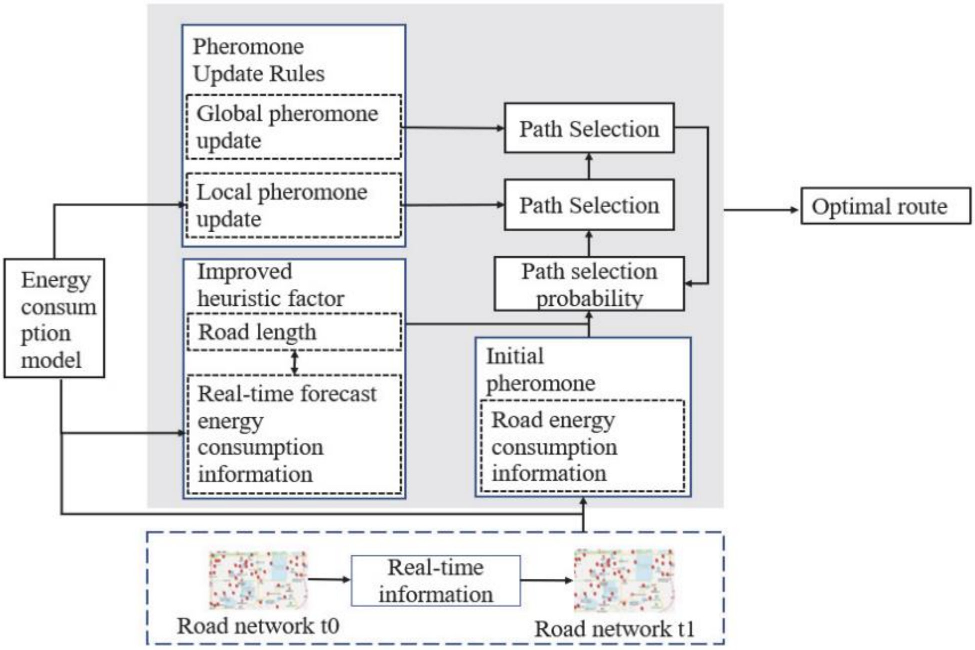

Figure 2.2 E-ACO Algorithm

In another case, there is a bilevel ACO that divides the problem into the capacitated vehicle routing problem and them fixed routing vehicle charging problem at the lower level, with different confidence thresholds that reject substandard service. To do this we adapt the thresholds to eject unsuitable service strings and instead concentrate on successful one (Jia et al., 2022b). Furthermore, enhancements in ACO for VRPs have been proposed to incorporate constraints where operations as the insertion of new solutions are designed and the convergence is made faster through selective rewarding of pheromone variables (Deng & Wu, 2023). The capacitated vehicle routing problem (CVRP) has also been solved using the enhanced ant colony system (EACS) algorithm and includes the K-nearest neighbour (KNN) algorithm in generating the initial solutions and sub path diversification techniques to avoid early convergence (Ahmed et al., 2023). These adaptive learning techniques enhance the facility of solving the CEVRP by the ACO algorithm and help in saving energy consumption and computational time and in addition providing better solutions. Applying these methods, it shows that ACO can learn and search the best route in an ever-changing environment of the eV routing problem.

2.5.1.2 Exploration and Exploitation Balance

Balancing exploration and exploitation are important when applying ACO algorithms and this is equally important when solving the CEVRP problem. This balance is worked out in the so-called E-ACO algorithm especially for dynamic energy efficient path planning of Electric Vehicles accommodating a traffic flow prediction model and an EV’s energy consumption model. Thus, the restructuring of heuristic factors and state transition rules improves both, time and accuracy of path planning in comparison with traditional ACO approaches (“Dynamic Energy-Efficient Path Planning for Electric Vehicles Using an

Enhanced Ant Colony Algorithm," 2024). Also, the EACS solving CEVRP applies the KNN algorithm to find the optimal initial solution and with its application of sub paths to improve the diversity mechanism; this appears to get rid of some problems such as the premature convergence and stagnation. This method therefore leads to riskier moves but on the overall there is a better balance between exploitation and exploration and the routes found within a reasonable amount of time all in real computational time (Ahmed et al., 2023). In addition, the vehicle routing problem with path, time window, and capacity constraints is solved by an enhanced Ant Colony Optimization which accelerates convergence because only the pheromone of the currently best constructed overall path is rewarded. Consequently, it reduces the overall cost of the delivery route, does not exclude customers, and produces reasonable solutions (Deng & Wu, 2023). Consequently, the real-case application of CEVRP based on the theory describes the applications of the fuzzy c-means clustering technique to group the demand points and ACO for selecting the best path within each group. Utilization of data from the real environment and GPS technology makes it possible to identify at once that a particular route deviates from the optimum and hence give the best possible route suggestions; besides, it makes it possible to balance exploration and exploitation in real-time situations (Yavşan & İlhan, 2022). In total, the use of ACO algorithms to deal with CEVRP advances how to combine exploration and exploitation to get efficient and effective routing strategies.

2.5.2. Flowchart for ACO in CEVRP

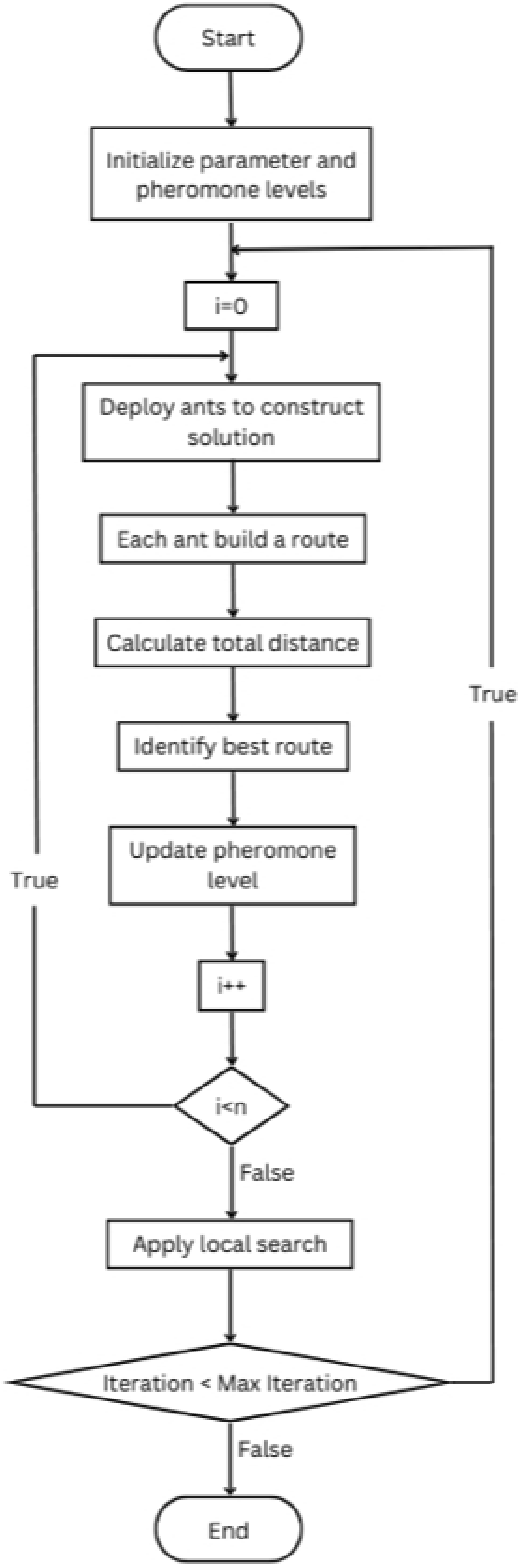

Figure 2.3 Flowchart for ACO in CEVRP (Jia et al., 2022a)

CHAPTER 3

METHODOLOGY

3.1 Introduction

The proposal of the methodology for this project describes the way applied to systematically solve the Capacitated Electric Vehicle Routing Problem (CEVRP) using the Ant Colony Optimization (ACO) algorithm. The CEVRP is an NP-hard problem of selecting the appropriate paths that the EVs should follow in delivering products to the customers, constrained by factors such as battery and vehicle capacity. The main goal is to minimize the overall distance traveled as well as to satisfy all the customers' needs and to return the EV to the depot. This paper describes preparation of the input data, setting up and optimisation of the ACO parameters, running of the algorithm, as well as evaluation of the outcomes. The initialization of the pheromone initiates the ACO process while the last step known as the local search process can be seen as the final stage of the algorithm since it helps to maximize the results of solving the CEVRP problem.

3.2 Problem Definition

The Capacitated Electric Vehicle Routing Problem (CEVRP) is an NP-hard problem that finds the optimal routes of a fleet of electric vehicles (EV) to satisfy a set of customers' demands while returning to the central depot. Every EV has its battery capacity that defines the total number of miles that can be traveled before recharging, and the load capacity to the total amount of load possible to carry by the EV. The objective of the CEVRP is to minimize the total distance traveled by the EVs while ensuring that the following constraints are met:

- **Demand Fulfilment:** Every customer must be served exactly once by the EV, and the total demand of the served customers cannot be more than the capacity of the EV.
    - ✓ **Maximum Vehicle Capacity Set in Current Case :** 100 units
- **Battery Constraints:** The distance to be covered by an EV from the depot and customers and the distance between two consecutive customers should not be more than the battery capacity of the EV. In this case, the EV can go to a charging station for recharging before it resumes the route it was following.
    - ✓ **Relationship between Battery and Distance :** 1 unit distance required 1 unit of battery
    - ✓ **Maximum Battery Capacity Set in Current Case :** 150 unit
- **Route Optimization:** The solution must determine which paths are shortening the overall distances of the EV's journey as much as possible and how to get the vehicle back to the depot by the end of the day.

The complexity of the CEVRP stems from the fact that these constraints must be met while trying to find the most optimal routes. Ordinary methods used for solving routing problems prove inefficient when complicating factors such as battery and load capabilities are added, more so when there are numerous customers and possible routes. This work seeks to tackle these challenges with the Ant Colony Optimization (ACO) algorithm, as a nature-inspired metaheuristic solution to efficiently search and find excellent near-optimal solutions in a timely manner. This kind of problem is suitable for the ACO algorithm since it can adjust different constrains and deliver good results in a reasonable time.

3.3 Data Preparation

Preparation of data is an important step when establishing the Ant Colony Optimization (ACO) algorithm to solve the Capacitated Electric Vehicle Routing Problem (CEVRP). This section identifies how the required data inputs in the ACO process were produced, formatted, and implemented.

3.3.1 Demand Data

The demand data is the amount of goods that a customer needs in the analysis of supply chain management. This dataset includes the following elements:

- Depot: The origin and terminal of EV routes that are associated with a demand value of zero.
- Customers: Every customer node has a specific need that has to be fulfilled by the EV. The demand values are derived to be a realistic order delivery demand and it is stored in the format of CSV file named as 'demand.csv' with each record in the CSV containing demand value of a particular customer.
- Charging Station: This is a node involving the charging place for the EVs if the batteries have been expended and they need to be charged once

more. The charging station similar to the depot has a demand value of zero indicating that vehicles will not be traveling to this location to fulfil demands.

The above demand data helps to avoid overloading the EVs while meeting all the customer demand rates.

	A	B
1	Node_ID	Demand
2	Depot	0
3	Customer_1	50
4	Customer_2	15
5	Customer_3	35
6	Customer_4	25
7	Customer_5	20
8	Customer_6	10
9	Customer_7	30
10	Customer_8	45
11	Customer_9	30
12	Customer_10	15
13	Charging_Station	0

Figure 3.1 Demand Data Set in Current Case

3.3.2 Distance Matrix

Another important matrix that is utilized to signify the distances between nodes is the distance matrix which contains distances for all the nodes even for the depot, customers, and the charging station. This matrix is structured as follows (Following current situation set, changeable for different situation) :

- 12x12 Matrix: Given that there are 10 customers, 1 depot, and 1 charging station, the distance matrix is a 12x12 matrix. Each element d_{ij} in the matrix represents the distance between node i and node j.

$$D = \begin{bmatrix} 0 & d_{01} & d_{02} & \cdots & d_{0n} \\ d_{10} & 0 & d_{12} & \ldots & d_{1n} \\ d_{20} & d_{21} & 0 & \ldots & d_{2n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ d_{n0} & d_{n1} & d_{n2} & \cdots & 0 \end{bmatrix} \tag{3.1}$$

  Where:

  d_{ij} represents the distance between node i and node j.

  d_{ii} =0 for all i, indicating no self-travel distance.

- Symmetric Matrix: Matrix is also said to be symmetric because the distance between node i and node j will be equal to the distance between node j and node i.
- Zero Diagonal: The diagonal components of the matrix are zero because the distance of any node to itself in the network is zero.

This matrix is saved in a CSV file with the name distance_matrix.csv and contains the information required by the ACO algorithm in calculate the total distance of each constructed route which is relevant when establishing the routes.

	A	B	C	D	E	F	G	H	I	J	K	L	M
1		Depot	Customer_1	Customer_2	Customer_3	Customer_4	Customer_5	Customer_6	Customer_7	Customer_8	Customer_9	Customer_10	Charging_Station
2	Depot	0	84	46	95	55	45	25	17	94	90	14	62
3	Customer_1	84	0	85	57	63	73	48	32	54	30	67	21
4	Customer_2	46	85	0	5	31	34	14	77	54	12	77	27
5	Customer_3	95	57	5	0	89	8	68	94	49	97	70	87
6	Customer_4	55	63	31	89	0	54	70	49	73	22	42	28
7	Customer_5	45	73	34	8	54	0	24	9	10	14	49	39
8	Customer_6	25	48	14	68	70	24	0	65	100	63	52	89
9	Customer_7	17	32	77	94	49	9	65	0	70	86	12	38
10	Customer_8	94	54	54	49	73	10	100	70	0	18	18	85
11	Customer_9	90	30	12	97	22	14	63	86	18	0	52	44
12	Customer_10	14	67	77	70	42	49	52	12	18	52	0	98
13	Charging_Station	62	21	27	87	28	39	89	38	85	44	98	0

Figure 3.2 Distance Matrix Data Set in Current Case

3.3.3 Loading and Validating Data

At the beginning of the ACO process the demand data and the distance matrix are loaded into it. To ensure that the input data is valid:

- Distance Validation: The distance matrix is then examined to avoid a situation whereby the distance between any two nodes in the path is more than the maximum battery range of the EVs. Mathematically, this condition can be expressed as:

$$d_{ij} \leq B \; for \; all \; i, j \qquad (3.2)$$

  Where B is the battery capacity of the EV. If any d_{ij} exceeds this capacity, the input is flagged as invalid, and the algorithm cannot proceed until the data is corrected.

Hence, through accurate preparation and validation of the demand data and distance matrix, a proper ground is laid to solve the CEVRP using the ACO algorithm. This data preparation clears all constraints used by the algorithm to solve the problem, thus leaving it to determine the best routes only.

3.4 Parameters Of the Ant Colony Optimization (ACO)

In the ACO algorithm, there are several parameters that can be viewed as critical since they significantly affect the ACO search process and the quality of the final solutions. These parameters determine how ants (virtual agents) search possible paths, when they switch between the exploration and exploitation of solutions, and when the algorithm becomes convergent to the near-optimal state. This section explains the parameters of the Ant Colony Optimization algorithms applied to this research work and explains their importance in addressing the Capacitated Electric Vehicle Routing Problem (CEVRP).

3.4.1 Number of Ants

The number of ants denote to the number of agents working at a time in each iteration of the algorithm to find the solution.

A numerous number of ants in each field size results to a diversification of the search since the ants explore more possibilities of pathways. On the other hand, it also leads to growth of the computational time. Both ants must not be too less or too more and there is a need to strike a mean to prevent high computational costs yet achieve good exploration.

3.4.2 Influence of Pheromone (α)

Here, the parameter α (alpha) determines the effect of the pheromone trails to the overall decision of the ants. Pheromones are called the chemical signals which ants lay on paths they are going and when there are higher levels of pheromone, they make a path more attractive to subsequent ants.

A higher α value focuses more on the reinforcement of the already learned good paths and has less exploration functionality, which results in faster convergence towards the optimal or near optimal solutions. Nonetheless, if α is too large, the algorithm is likely to converge too quickly to a local optimum because of reliance on pheromone trails. α must be balanced in order to achieve a good mix between exploitation, on finding better solutions and exploration in finding good paths.

3.4.3 Influence of Heuristic Information (β)

The parameter β (beta) defines the impact of the heuristic information which is the inverse of distance between nodes on the decisions made by the ants. This heuristic persuades ants to go for shorter trails.

A higher β value makes the algorithm greedier, favouring shorter routes even more than it already does. This can result in more efficient solutions, but it may also disadvantage the search by eliminating options that may take slightly longer to traverse an individual segment but are shorter overall. Therefore, adapting β enables a better control of the distance factor with relation to ratios of pheromone levels.

3.4.4 Pheromone Evaporation Rate (ρ)

The parameter ρ gives the evaporation rate of the pheromone on the paths. This gradual reduction in pheromone strength over time also serves to minimize the influence of older and maybe suboptimal paths in the searching process.

The evaporation rate is of particular importance when it comes to ensuring the diversity in the search space. A low ρ value indicates that pheromones have a slow rate of evaporation, meaning older paths can have more influence than is desirable, thus resulting in premature convergence. On the other hand, a high ρ value evokes fast degradation of the emitted pheromones which leads to more exploration of the search space but may hinder the algorithm to be more exploitive in the best-found solutions. It is, therefore, important to strike a identifying the right evaporation rate for the loop to be effective in balancing exploration and exploitation.

3.4.5 Pheromone Deposit Factor (Q)

The parameter Q influences the quantity of the pheromone on a path when an ant finishes a particular route. The quantity of pheromone that gets deposited increases with the quality of the solution, so the better solutions lead to greater deposition of pheromone.

Larger Q value enhances the reinforcement of good solutions hence making the paths associated with these solutions preferable to the subsequent ants. However, if Q is too high, it is going to reinforce one path a little too much, thus lowering the diversification in the search. The Q values are balanced in such a way that good solutions get enough reward while at the same time the search is not hampered by this reward.

All these parameters are quite important to the working of the ACO algorithm. All these parameters affect the behaviour of the ants, the nature of the search, and how close the algorithm gets to a global or near-global optimum. Therefore, it is crucial to fine-tune these parameters in order to solve the CEVRP and achieve a good balance between exploration and exploitation in the algorithm, as well as preventing it from getting stuck in local optima, while at the same time finding the best routes according to the problem constraints such as battery capacity and load of the vehicles.

3.5 Initialization

In this study, the initialization phase plays a very significant role as it precedes the optimization phase that aims at solving the Capacitated Electric Vehicle Routing Problem (CEVRP) using the Ant Colony Optimization (ACO) algorithm. In this phase some of the components of algorithm such as the pheromone matrix, initial variables and conditions, as well as validation criteria are made. It helps the algorithm start searching for the best solution with a proper framework set in place.

3.5.1 Pheromone Matrix Initialization

The pheromone matrix is an important part of ACO algorithm, where pheromone amount describes pheromone that exists on each path between nodes, such as the depot, customers and the charging station.

To begin with the algorithm, the intensity of pheromone in all the paths are equivalent and are set to a small and fixed value. This uniform initialization is because there is no prior information about which paths will yield the best solutions.

$$T_{ij}(0) = T_0 \quad \text{for all } (i, j) \tag{3.3}$$

Where:

- $T_{ij}(0)$ is the initial pheromone level on the path between node i and node j.
- T_0 is a small constant, typically chosen based on the problem scale.

Initializing each of the pheromone matrices with the same values ensures that the algorithm begins its search for solutions equally distributed within the solution space. To achieve this, pheromone levels will be changed as the algorithm moves forward; this will help to direct the ants in the direction of the better solutions.

3.5.2 Ant Initialization

Ants as Agents: The Ant Colony Optimization (ACO) algorithm used in this paper has ants as the working agents that represent the search for various routes. At the start of the process, several ants are generated, each of which will correspond to a possible solution to the problem.

Starting Position: Every travelling ants starts from the depot (node 0). From here, they would get to different locations of all customers taking into consideration factors such as battery capacity and vehicle load limit.

Parameters: Here the number of ants is defined as one of the components of the ACO parameters and each of them is acting according to the initial pheromone deposited with relative heuristic information (distances).

3.5.3 Validation of Input Data

Distance Validation: First check is performed on the distance matrix before applying the Ant Colony Optimization (ACO) algorithm where no distance between any two nodes is greater than the maximum battery capacity of the electric vehicle (EV). This check is important because if an EV cannot travel from one node to another without running out of battery, then the solution is impossible.

$$d_{ij} \leq B \quad \text{for all } i, j \tag{3.4}$$

Where:

- B is the maximum battery capacity of the EV.

Demand Validation: The demand data is also validated to ensure that the total demand assigned to the EV shall not exceed the load capacity of the EV. This makes sure that every customer requirement can be fulfilled within the ability of the EV.

Charging Station Identification: The initialization step also concerns with defining the charging station in the data that it must be well-defined as the node that EV can be recharged. This is helpful in directing EV that still have other customers to attend to but are almost out of charge.

3.5.4 Initialization of Other Variables

Tracking Variables: Variables for recording the best solution found, total distances traveled, and the battery levels are created. These variables will be modified throughout the run of the algorithm as well as throughout the discovery of the different paths by the ants.

Randomization: Due to possible bias, some of the aspects of the initialization may be randomly selected, for example, the initial selection of the first customer node by an ant. It also helps when looking for a diverse number of solutions in the initial steps of the algorithm's execution.

The initialization phase basically sets up the pheromone matrix for the ACO algorithm, prepares the ants, and checks the input data for validity. These steps make it possible for the algorithms, at least, start from a fair state in the hyperspace so that the search can proceed across the hyper space and result into possible optimal or near-optimal solutions to the CEVRP. Great care must be taken when initializing the algorithm because it affects the quality of solutions returned by the algorithm and the speed at which the solutions are found.

3.6 Ant Colony Optimization (ACO) Process

The main algorithmic step in the whole the algorithm is the ACO process, in which artificial ants construct the solutions, deposit the pheromones and update them after each iteration for reaching the optimum or near optimal solution for the Capacitated Electric Vehicle Routing Problem (CEVRP). This section describes the ACO steps, pointing out how the algorithm influences and adapts the problem solution to find the best routes satisfying all requirements.

3.6.1 Ant Deployment and Route Construction

Initial Deployment: At the beginning of each iteration, the specific number of ants is released from the depot (start node). Every ant corresponds to a feasible solution to the routing problem. These ants then start constructing their paths by selecting the subsequent customer to move to, according to the level of pheromones and heuristic data (e.g. the inverse of distance).

Transition Probability: The selection of which customer to visit next is done based on the probability, which consider the amount of pheromone and heuristic. The probability P_{ij} that an ant will move from node i to node j is given by:

$$P_{ij} = \frac{(T_{ij})^{\alpha} \cdot (n_{ij})^{\beta}}{\sum_{k \epsilon allowed} (T_{ik})^{\alpha} \cdot (n_{ik})^{\beta}} \tag{3.5}$$

(Duan et al., 2015)

Where:

- T_{ij} is the pheromone level on the path from node i to node j.
- $n_{ij} = \frac{1}{d_{ij}}$ is the heuristic information (inverse of distance).
- α and β are parameters controlling the influence of pheromone and heuristic information respectively.
- The denominator sums over all possible next nodes that the ant has not yet visited.

Battery and Load Constraints: While building their routes, ants always ensure that the remaining battery capacity can take them to the next customer or the charging station, or the vehicle load can meet the demand at the next customer. If the battery is not enough to get to the next node, the ant will visit to the charging station.

Route Completion: Every ant is still constructing its route until all the customers have been serviced, and the ant goes back to the depot. If an optimal solution is not found on the route due to battery or load limitation, then this route is rejected, and that ant will not have any role to make any change in the pheromone trail.

3.6.2 Pheromone Update

After all ants have completed their routes, the pheromone levels on the paths are updated to reflect the quality of the solutions found.

Pheromone Evaporation: First, all the pheromone present on all paths evaporates by a factor of ρ (the evaporation rate). This prevents pheromone levels from accumulating indefinitely and encourages exploration of new paths:

$$T_{ij} \leftarrow (1-\rho) \cdot T_{ij} \text{ (Hutter, n.d.).} \quad (3.6)$$

Where:

- ρ is the evaporation rate, typically a small positive number.

Pheromone Deposition: Following evaporation, only the paths that belong to the best solutions that have been discovered in the current iteration are subjected to further pheromone deposits. The amount of pheromone deposited on a path is inversely proportional to the total distance of the route:

$$T_{ij} \leftarrow T_{ij} + \sum_{ants} \frac{Q}{L_{ant}} \tag{3.7}$$

Where:

- Q is the pheromone deposit factor.
- L_{ant} is the total distance of the route taken by the ant.
- The summation is over all ants that used the path from node i to node j in their routes.

3.6.3 Best Route Identification

After a change in the pheromone levels, the algorithm determines the shortest route detected during the current cycle. This route is normally that which has the minimum total distance that also satisfies all the problem constraints (all customer visits, depot return and battery plus load constraints).

Global Best Update: However, if the best route produced in the present iteration is shorter than the global best route then, the global best route will be updated to this optimal route.

3.6.4 Convergence and Iteration Control

The ACO process continuously goes through ants' deployment, route construction and pheromone updating until the following conditions is reached.

Maximum Iterations: The algorithm iterates until a specified number of iterations have been completed (max_iterations). Once it does that, the best solution reached during the execution is returned as the final outcome.

3.6.5 Local Search

To enhance the developed solutions even more, the local search methods for example 2-opt swap, can be used with the best routes found in a period by the ants. This involves changing the route a little, for example, by swapping one node with another to check whether a shorter path exists.

ACO process involves sending ants to build solutions that are possible, using their perceptions to update the quality of pheromones and proceeding with the search again and again until finding the best possible solution. By enabling of the exploration and exploitation stages, where the exploration stage is facilitated by probabilistic decision-making and pheromone evaporation, and the exploitation stage is realized by pheromone reinforcement, the algorithm achieves to solve the CEVRP problem to identify routes which yield minimum total distance while at the same time satisfying all the constraints.

CHAPTER 4

ANALYSIS OF RESULT

4.1 Introduction

In the analysis of the results, we mainly pay attention to the assessment of the ACO algorithm for solving the CEVRP problem. This involves analysing most notably the total distance of the best routes found, time taken as well as the convergence patterns of the algorithm used. When examining these outcomes, our goal is to investigate how well ACO is likely to solve a routing problem with the imposed constraints and determine which specific aspects of the problem the algorithm works most efficiently on. This analysis gives specific information on the effectiveness and functionality of achieving ACO implementation, thus affirming the chosen strategy.

4.2 Taguchi Design for ACO's Parameters

Taguchi design is a statistical methodology developed by Genichi Taguchi of Japan aimed at raising quality of products that are produced. The Taguchi design is a statistical method used to optimize process parameters to enhance quality by carrying out studies on several factors at one time with little experimentation. In the present case, while using ACO, the Taguchi design was used appropriately to optimize parameters including the number of ants, the influence of pheromone (α), influence of distance (β), evaporation rate (ρ), and pheromone deposit factors (Q). To find out the values of these parameters that will significantly reduce the total distance of the best route but will not compromise the program's effectiveness, we conducted experiments by varying these parameters using an orthogonal array. This systematic approach helped in determining how each parameter impacted on the ACO algorithm and consequently provide an efficient solution towards Capacitated Electric Vehicle Routing Problem (CEVRP).

4.2.1 Taguchi Setting

When applied to Ant Colony Optimization (ACO) algorithms, Taguchi method allows for the systematic optimization of ACO parameters, leading to improved credibility and accuracy in the solutions. To refine the parameters of ACO, the parameters are set as shown in Table 4.1.

Table 4.1 Levels and Factors for Taguchi Analysis

Levels	Factors				
	Number of Ants	Influence of pheromone (α)	Influence of distance (β)	Evaporation rate (ρ)	Pheromone deposit factors (Q)
1	10	1	1	0.1	10
2	30	3	3	0.4	100
3	50	5	5	0.7	1000

This analysis involves 5 factors with 3 levels that are number of ants, influence of pheromone, influence of distance, exploration rate, and pheromone deposit factors. The Taguchi method enables efficient exploration of the influence of these factors on the performance of parameters exhaustively. The orthogonal array selected for this experiment is $L_{27}(3^5)$, L27 means that the orthogonal array contains 27 experiments, (3^5 indicates that there are 5 factors, each of which can take on 3 levels. All the predefined tests, as shown in Table 4.2, were executed to solve the benchmark problem.

Table 4.2 Number of Test for Different Combinations of Factors

No. of Test	Factors				
	Number of Ants (ant_num)	Influence of pheromone (α)	Influence of distance (β)	Evaporation rate (ρ)	Pheromone deposit factors (Q)
1	10	1	1	0.1	10
2	10	1	1	0.1	100
3	10	1	1	0.1	1000
4	10	3	3	0.4	10
5	10	3	3	0.4	100
6	10	3	3	0.4	1000
7	10	5	5	0.7	10
8	10	5	5	0.7	100
9	10	5	5	0.7	1000
10	30	1	3	0.7	10
11	30	1	3	0.7	100

12	30	1	3	0.7	1000
13	30	3	5	0.1	10
14	30	3	5	0.1	100
15	30	3	5	0.1	1000
16	30	5	1	0.4	10
17	30	5	1	0.4	100
18	30	5	1	0.4	1000
19	50	1	5	0.4	10
20	50	1	5	0.4	100
21	50	1	5	0.4	1000
22	50	3	1	0.7	10
23	50	3	1	0.7	100
24	50	3	1	0.7	1000
25	50	5	3	0.1	10
26	50	5	3	0.1	100
27	50	5	3	0.1	1000

This table shows combination of the identified important parameters related to the Ant Colony Optimization (ACO) algorithm applying the Taguchi method. It is composed of 27 test scenarios differing in number of ants, the influence of pheromone (α), the influence of distance (β), the evaporation rate (ρ), the pheromone deposit factor (Q). The idea is to identify how these various permutations affect the ACO algorithm when employing it to solve the Capacitated Electric Vehicle Routing Problem(CEVRP). This approach can potentially explore different parameter settings, thereby determining which configuration will yield the best results in terms of algorithm speed and solution quality.

Table 4.3 Final Distance as Responses for the Problem of each Test

No. of Test	Response	
	Final Route	Unit Distance
1	[0, 3, 11, 1, 10, 0, 4, 2, 9, 11, 7, 0, 5, 8, 0, 6, 0]	680
2	[0, 3, 11, 1, 10, 0, 4, 2, 9, 11, 7, 0, 5, 8, 0, 6, 0]	680
3	[0, 3, 11, 1, 10, 0, 4, 2, 9, 11, 7, 0, 5, 8, 0, 6, 0]	680
4	[0, 7, 5, 3, 2, 0, 10, 8, 9, 0, 6, 4, 0, 1, 11, 0]	542
5	[0, 7, 5, 3, 2, 0, 10, 8, 9, 0, 6, 4, 0, 1, 11, 0]	542
6	[0, 7, 5, 3, 2, 0, 10, 8, 9, 0, 6, 4, 0, 1, 11, 0]	542
7	[0, 7, 5, 3, 2, 0, 10, 8, 9, 0, 6, 1, 11, 4, 0]	402
8	[0, 7, 5, 3, 2, 0, 10, 8, 9, 0, 6, 1, 11, 4, 0]	402
9	[0, 7, 5, 3, 2, 0, 10, 8, 9, 0, 6, 1, 11, 4, 0]	402
10	[0, 6, 2, 3, 5, 0, 10, 8, 9, 0, 7, 4, 0, 1, 11, 0]	525
11	[0, 6, 2, 3, 5, 0, 10, 8, 9, 0, 7, 4, 0, 1, 11, 0]	525
12	[0, 6, 2, 3, 5, 0, 10, 8, 9, 0, 7, 4, 0, 1, 11, 0]	525
13	[0, 7, 5, 3, 2, 0, 10, 8, 9, 0, 6, 1, 11, 4, 0]	402
14	[0, 7, 5, 3, 2, 0, 10, 8, 9, 0, 6, 1, 11, 4, 0]	402

15	[0, 7, 5, 3, 2, 0, 10, 8, 9, 0, 6, 1, 11, 4, 0]	402
16	[0, 6, 9, 8, 11, 0, 7, 5, 3, 2, 0, 10, 4, 1, 11, 0]	540
17	[0, 6, 9, 8, 11, 0, 7, 5, 3, 2, 0, 10, 4, 1, 11, 0]	540
18	[0, 6, 9, 8, 11, 0, 7, 5, 3, 2, 0, 10, 4, 1, 11, 0]	540
19	[0, 7, 5, 3, 2, 0, 10, 8, 9, 0, 6, 1, 11, 4, 0]	402
20	[0, 7, 5, 3, 2, 0, 10, 8, 9, 0, 6, 1, 11, 4, 0]	402
21	[0, 7, 5, 3, 2, 0, 10, 8, 9, 0, 6, 1, 11, 4, 0]	402
22	[0, 6, 9, 8, 11, 0, 7, 5, 3, 2, 0, 10, 4, 1, 11, 0]	540
23	[0, 6, 9, 8, 11, 0, 7, 5, 3, 2, 0, 10, 4, 1, 11, 0]	540
24	[0, 6, 9, 8, 11, 0, 7, 5, 3, 2, 0, 10, 4, 1, 11, 0]	540
25	[0, 7, 5, 3, 2, 0, 10, 8, 9, 0, 6, 1, 11, 4, 0]	402
26	[0, 7, 5, 3, 2, 0, 10, 8, 9, 0, 6, 1, 11, 4, 0]	402
27	[0, 7, 5, 3, 2, 0, 10, 8, 9, 0, 6, 1, 11, 4, 0]	402

4.2.2 Taguchi Analysis

Signal-to-Noise (S/N) ratio is one of the primary factors in the Taguchi method used to determine the quality level and design stability of a process. Specifically, in the field of the present project, which is the study of Ant Colony Optimization (ACO) algorithm parameter settings, S/N ratio is the primary indicator to find the proper settings that provide the optimal results consistently. More precisely, the project has adopted the objective of reducing the total distance that is covered by the Electric Vehicles (EVs) in a capacitated vehicle routing problem. Thus, by using the "Smaller-the-Better" S/N ratio, this analysis is aimed at minimizing the average distance and, across multiple trials, the variability in improving the reliability and effectiveness of the ACO algorithm. The S/N Ration is calculated using the formula:

$$\text{S / N Ratio} = -10 \times \log_{10}\left(\frac{1}{n}\sum_{i=1}^{n} y_i^2\right) \tag{4.1}$$

Where:

- y_i represents the observed value of the response variable (total route distance) for the ith trial in this case.
- n is the number of trials or observations.

In this project, the response variable denoted by y_i represents the total route distance estimated by the ACO algorithm when the input parameters are set at a certain level of ant_num, α, β, ρ and Q. It is important to identify the values of these parameters that would not only yield the lowest average Route Distance but also the lowest standard deviation of the results in different simulation runs.

The Smaller-the-Better S/N ratio is of negative logarithmic form hence greater values of the response variable (short route distances) imply higher S/N ratios which are more desirable. As a result, the increase in the S/N ratio reveals the optimal settings for the examined parameters, which yield the smallest and most constant route distance. This approach guarantees that the ACO algorithm will work effectively, providing near-optimal routes all the time, because of the random probability distribution in the ACO algorithm.

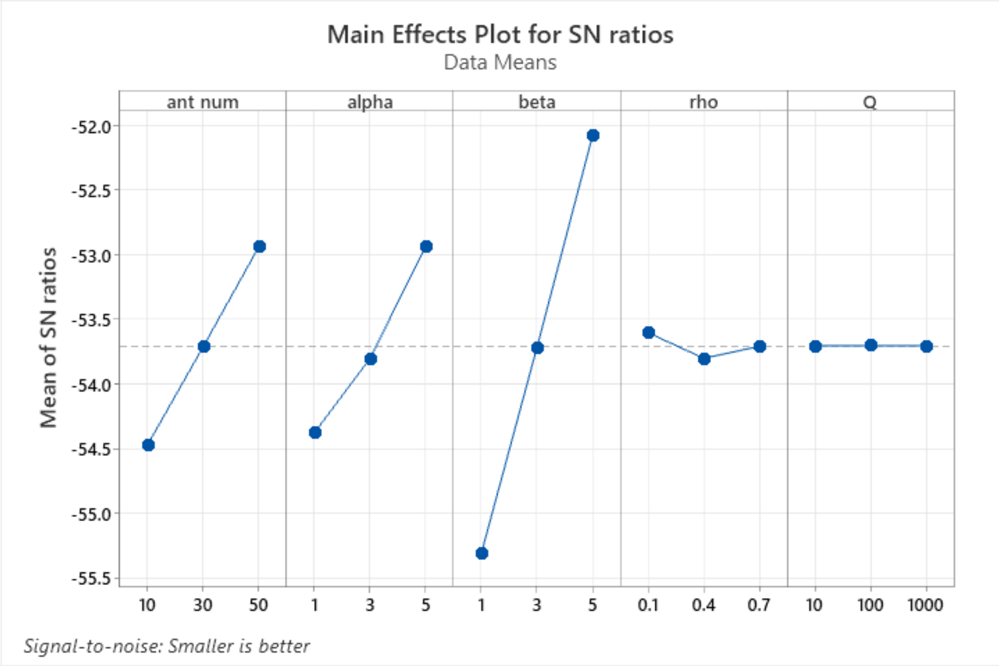

Figure 4.1 S/N Ratio Plot for Problem

The plot of the S/N ratios shows the relationship between the elements of the factors: ant_num, alpha, beta, rho and Q and their influence on the total route distance in terms of stability.

Ant_num (Number of Ants) : The plot shows that as the number of ants is increased from 10 to 50, the S/N ratio has increased implying that a greater number of ants makes the algorithm more robust. This may be since the solutions have become more diverse leading to the consistent discovery of shorter solutions.

Alpha, α (Pheromone Influence) : A low value of the S/N ratio is observed when alpha is set at 1 while the ratio gradually increases as alpha increases up to 5. These trends suggest that increasing the pheromone influence enhances the stability of ant route optimization because ants are better at utilizing previously identified best routes during subsequent iterations.

Beta, β (Heuristic Information Influence) : The S/N ratio increases sharply as beta increases from 1 to 5 and reaches its highest value at beta = 5. This implies that an increased focus on heuristic information, in this case the ‘’inverse distance’’, enhances the stability and efficiency of the algorithm in terms of the overall routes distance.

Rho, ρ (Pheromone Evaporation Rate) : The S/N ratio does not appear to greatly vary with different rho levels and has a slight decrease at rho = 0.1. This stability indicates that the rate of pheromone evaporation is less effective for the route distance consistency, demonstrating insensitiveness to changes in the parameter.

Q (Pheromone Deposit Factor) : The plot demonstrates that the S/N ratio remains constant for each level of Q, which means that the volume of the deposited pheromone does not influence the stability of the route distance in this context. It could mean that after depositing a certain amount of pheromone, any additional amount is ineffective.

The main effects plot of S/N ratios showing that out of all the parameters, beta (heuristic information influence) and ant_num (number of ants) have the most influence on the ACO’s stability where larger values lead to improved and consistent route distances. On the other hand, the parameters rho, which controls the evaporation rate of the pheromone, and Q which is related to the deposit factor of pheromone, impacts on the stability rather low, indicating that the algorithm has low sensitivity to the variations in these parameters. By identifying the parameter settings that improves the S/N ratio of the solution, the project can improve the stability of the ACO algorithm in return, the algorithm should be able to locate the shortest route for each vehicle with little deviation in the result.

Table 4.4 Optimal Parameter Settings and Their Ranks of Significance for Problem

	Factor				
	Number of Ants	Influence of pheromone (α)	Influence of distance (β)	Evaporation rate (ρ)	Pheromone deposit factors (Q)
Optimal Value	50	5	5	0.1	10/100/1000
Rank	2	3	1	4	5

4.2.3. Analysis of Variance for S/N Ratio

The Analysis of Variance (ANOVA) result for the S/N ratios across different problems help to understand the statistical significance of the parameters in terms of their impact on the S/N ratios. The results obtained are provided in Table 4.5.

Table 4.5 P-value from ANOVA for S/N ratios

Factors	P-value
Number of Ants	0.00
Influence of pheromone (α)	0.00
Influence of distance (β)	0.00
Evaporation rate (ρ)	0.00
Pheromone deposit factors (Q)	0.39

In the context of Taguchi method, the significance level is typically set as 0.05. This means that when conducting an ANOVA analysis, any p-value below 0.05 indicates that the factor is statistically significant and has a meaningful impact on the outcome. In this case, the P-values derived from the analysis suggest that the parameters number of ants, influence of pheromone (α), influence of distance (β) and evaporation rate (ρ) all have P-values equals to 0. 00, which means that these factors play a significant role in determining the total distance of the route. This result indicates that variations of these factors resulting in a statistically significant impact on the ACO algorithm performance reemphasizes the importance of these aspects in enhancing the route efficiency. Consequently, these parameters should be optimized in a manner that will produce the best outcomes in terms of minimizing the route distance.

On the other hand, the P-value for pheromone deposit factor Q was calculated to be 0. 39, which exceeds the significance level of 0. 05. This means that Q is not significant at this level, which implies that changes in the amount of pheromone deposited does not significantly affect the total route distance. Such inference means that once the pheromone trail is established with a certain degree of quantity, further enhancements in the parameter Q do not help in advancing the performance of the algorithm. Therefore, further tuning on the less influential parameters like Q should not be given much attention as they have minimal impact in this setup, but instead, should normally focus on increasing parameters like number of ants, α, β, and ρ.

Table 4.6 Final Route and Total Distance using Optimal Parameter

Optimal Route	Total unit Distance
[0, 7, 5, 3, 2, 0, 10, 8, 9, 0, 6, 1, 11, 4, 0]	402

From the optimal obtain in table 4.6, the electric vehicle is suggested to leave the depot at the beginning and visit to customer 7, followed by customer 5, customer 3 and customer 2. After visiting customer 2, the electric vehicle will back to the depot for demand refill and battery recharging. After leaving the depot, the electric vehicle is suggested to visit to customer 10, customer 8. customer 9 and then back to the depot for the demand refill and battery recharge. The electric vehicle are next visiting the customer 6 and customer 1. After visiting the customer 1, the electric vehicle was suggested to visit to the charging station for battery charging. After battery recharged, the electric vehicle was suggested to visit customer 4 who is the last customer and finally back to the depot. With a total shorten distance which is 402 unit distance

CHAPTER 5

CONCLUSION AND RECOMMENDATION

5.1 Limitations

The first limitation of this project is that the problem and solution space are complex and variable when employing the Ant Colony Optimization (ACO) algorithm. ACO algorithm is very effective and versatile, but at the same time requires careful tuning of the parameters such as ant_num, α, β, ρ, and Q. Therefore, the ability to provide relatively stable results is sometimes challenging due to fluctuations in performance, especially when facing large numbers of constraints in the routes during search.

Another considerable limitation is that, in the routing scenario, only one charging station has been incorporated. This constraint requires the electric vehicle to always return to the same charging station whenever it needs to be recharged which may cause relatively long distances in larger or distributed networks. Also, the total distance optimization approach adopted from the project may lack consideration of other aspects like computational complexity or variation in the planning environment. Another obvious constraint is that in solving problems, all the parameters related to the problem are assumed constant and easily determined in advance, while in practice at any given time conditions may be vary.

5.2 Recommendations for Addressing Limitations

To overcome such limitations, it is suggested that future work should extend the study parameter space using advanced approaches such as machine learning or adaptive algorithm to self-tune the parameter in response to the variation in the problem domain. The addition of real time data and self-adaptive mechanisms add to the solidity of the algorithm and make it more suitable for dynamic environments which are characterized by uncertainties.

Furthermore, adding more than one charging station into the routing scenario will give the electric vehicles more charging opportunities thereby minimizing the total route distance and hence increase efficiency of the algorithm. This modification will enable the algorithm to consider other more accurate conditions that depicts logistics environment, thereby makes the algorithm more practical in real-life operations.

Besides, going beyond the objective function to accommodate other objectives for optimization by including other criteria such as computation time, changes in demand and flexibility of constraints would come up with a more comprehensive optimization of routes. Last but not the least, more investigation for a broader range of problems and would also further prove the viability of the proposed algorithm.

5.3 Achievement of Research Objectives

The aim of this study was to find the best solution to the total route distance on the CEVRP using ACO algorithm. The project of parameter optimization implemented using Taguchi method and ANOVA analysis in a systematic manner was able to achieve the goal of identifying the important parameters such as number of ants, influence of pheromone (α), influence of distance (β) and evaporation rate (ρ) affecting the performance of the algorithm. The results suggest that all these parameters are essential in reducing the route distance as it works towards achieving the goal of the project. Nevertheless, the project also showed that contribution of the pheromone deposit factors (Q) to the result is significantly low, and therefore, further optimization should be made on the parameters which are more influential.

5.4 Conclusion

In conclusion, this paper managed to apply and implement the Taguchi method and ANOVA analysis in enhancing the optimization of the (ACO) algorithm for the Capacitated Electric Vehicle Routing Problem (CEVRP) in minimizing the total distance for a route while fulfilling all the customer demands. However, while successfully attaining the primary goal of the study, the project also highlighted several weaknesses mainly, concerning the stability of the algorithm in choice of parameters and flexibility to changing contexts. As for future work, the presented ACO algorithm should be enforced with adaptive structures, more innovative optimization goals should be included, and the method should be tested in more diverse and realistic environments. Through overcoming these limitations, the ACO algorithmically can be enhanced to offer better and more accurate solutions of real-life routing issues in the future.

REFERENCE

Ahmed, Z. H., Hameed, A. S., Mutar, M. L., & Haron, H. (2023). An Enhanced Ant Colony System Algorithm Based on Subpaths for Solving the Capacitated Vehicle Routing Problem. *Symmetry*, *15*(11). https://doi.org/10.3390/sym15112020

Alif, A., Garside, A. K., Amallynda, I., & Ramadhani, B. N. I. F. (2022). Solving Capacitated Vehicle Routing Problem Using Football Game Algorithm. *Jurnal Ilmiah Teknik Industri*, *21*(1), 27–36. https://doi.org/10.23917/jiti.v21i1.15812

Amiri, A. (n.d.). *ELECTRIC VEHICLE ROUTING PROBLEM AND SOLUTION APPROACHES*.

Anugraha, R. A., Wiraditya, M. Y., Iqbal, M., & Darmawan, N. M. (2019). Application of Taguchi Method for Optimization of Parameter in Improving Soybean Cracking Process on Dry Process of tempeh Production. *IOP Conference Series: Materials Science and Engineering*, *528*(1), 012070. https://doi.org/10.1088/1757-899X/528/1/012070

Asghari, M., & Al-e-hashem, S. M. J. M. (2021). Green vehicle routing problem: A state-of-the-art review. *International Journal of Production Economics*, *231*, 107899. https://doi.org/10.1016/j.ijpe.2020.107899

Barak, M. (2013). Impacts of learning inventive problem-solving principles: Students' transition from systematic searching to heuristic problem solving. *Instructional Science*, *41*(4), 657–679. https://doi.org/10.1007/s11251-012-9250-5

Benotmane, Z., Kaoutar, G., Aoued, S., & Belalem, G. (2024). A New Resolution Strategy for Dynamic Capacitated Vehicle Routing Problem with Time Window. In M. S. Uddin & J. C. Bansal (Eds.), *Proceedings of International Joint Conference on Advances in Computational Intelligence* (pp. 407–419). Springer Nature Singapore.

Bezzi, D., Ceselli, A., & Righini, G. (2023). A route-based algorithm for the electric vehicle routing problem with multiple technologies. *Transportation Research Part C: Emerging Technologies*, *157*, 104374. https://doi.org/10.1016/j.trc.2023.104374

Carmen Barletta, C. T., Wolfgang Garn, & Fallah, S. (2023). Hybrid fleet capacitated vehicle routing problem with flexible Monte–Carlo Tree search. *International Journal of Systems Science: Operations & Logistics*, *10*(1), 2102265. https://doi.org/10.1080/23302674.2022.2102265

Chanachan, C., Sirisethakarn, M., Khapla, P., & Polvichai, J. (2023). A Comparison of Route Optimization Algorithms on Capacitated Vehicle Routing Problem. *2023 International Technical Conference on Circuits/Systems, Computers, and Communications (ITC-CSCC)*, 1–6. https://doi.org/10.1109/ITC-CSCC58803.2023.10212815

Chen, W., Shao, Z., Wang, K., Chen, X., & Biegler, L. T. (2010). Convergence depth control for interior point methods. *AIChE Journal*, *56*(12), 3146–3161. https://doi.org/10.1002/aic.12225

Choudhari, A., Ekbote, A., & Chaudhuri, P. (n.d.). *Capacitated Vehicle Routing Problem Using Conventional and Approximation Method*.

Cokyasar, T., Subramanyam, A., Larson, J., Stinson, M., & Sahin, O. (2023). Time-Constrained Capacitated Vehicle Routing Problem in Urban E-Commerce Delivery. *Transportation Research Record: Journal of the Transportation Research Board*, *2677*(2), 190–203. https://doi.org/10.1177/03611981221124592

Davatgari, A., Cokyasar, T., Subramanyam, A., Larson, J., & Mohammadian, A. (Kouros). (2024). Electric vehicle supply equipment location and capacity allocation for fixed-route networks. *European Journal of Operational Research*, *317*(3), 953–966. https://doi.org/10.1016/j.ejor.2024.04.022

De Andoin, M. G., Bottarelli, A., Schmitt, S., Oregi, I., Hauke, P., & Sanz, M. (2023). Formulation of the Electric Vehicle Charging and Routing Problem for a Hybrid Quantum-Classical Search Space Reduction Heuristic. *2023 IEEE 26th International Conference on Intelligent Transportation Systems (ITSC)*, 5318–5323. https://doi.org/10.1109/ITSC57777.2023.10421787

de Vos, M. H., van Lieshout, R. N., & Dollevoet, T. (2024). Electric Vehicle Scheduling in Public Transit with Capacitated Charging Stations. *Transportation Science*, *58*(2), 279–294. https://doi.org/10.1287/trsc.2022.0253

Deng, H., & Wu, J. (2023). Study on Optimization of Vehicle Routing Problem by Improved Ant Colony Algorithm. *2023 China Automation Congress (CAC)*, 399–404. https://doi.org/10.1109/CAC59555.2023.10450625

Duan, Z., Sun, S., Sun, S., & Li, W. (2015). Stochastic time-dependent vehicle routing problem: Mathematical models and ant colony algorithm. *Advances in Mechanical Engineering*, *7*(11), 168781401561863. https://doi.org/10.1177/1687814015618631

Dynamic Energy-Efficient Path Planning for Electric Vehicles Using an Enhanced Ant Colony Algorithm. (2024). *Tehnicki Vjesnik - Technical Gazette*, *31*(2). https://doi.org/10.17559/TV-20230510000618

Fazeli, S. S., Venkatachalam, S., & Smereka, J. M. (2024). Efficient algorithms for electric vehicles' min-max routing problem. *Sustainable Operations and Computers*, *5*, 15–28. https://doi.org/10.1016/j.susoc.2023.07.002

Froger, A., Jabali, O., Mendoza, J. E., & Laporte, G. (2022). The Electric Vehicle Routing Problem with Capacitated Charging Stations. *Transportation Science*, *56*(2), 460–482. https://doi.org/10.1287/trsc.2021.1111

Gong, I., Lee, K., Kim, J., Min, Y., & Shin, K. (2020). Optimizing Vehicle Routing for Simultaneous Delivery and Pick-Up Considering Reusable Transporting Containers: Case of Convenience Stores. *Applied Sciences*, *10*(12), 4162. https://doi.org/10.3390/app10124162

Guo, J., Wu, Z., & Wu, Z. (2008). An improved particle swarm optimization algorithm based on velocity updating. *2008 IEEE Conference on*

Cybernetics and Intelligent Systems, 330–334. https://doi.org/10.1109/ICCIS.2008.4670962

Hutter, F. (n.d.). *SLS-DS 2009: Doctoral Symposium on Engineering Stochastic Local Search Algorithms*.

Jia, Y.-H., Mei, Y., & Zhang, M. (2022a). A Bilevel Ant Colony Optimization Algorithm for Capacitated Electric Vehicle Routing Problem. *IEEE Transactions on Cybernetics*, *52*(10), 10855–10868. https://doi.org/10.1109/TCYB.2021.3069942

Jia, Y.-H., Mei, Y., & Zhang, M. (2022b). Confidence-Based Ant Colony Optimization for Capacitated Electric Vehicle Routing Problem With Comparison of Different Encoding Schemes. *IEEE Transactions on Evolutionary Computation*, *26*(6), 1394–1408. https://doi.org/10.1109/TEVC.2022.3144142

Jiang, H., Zhang, J., Xuan, J., Ren, Z., & Hu, Y. (2010). A Hybrid ACO algorithm for the Next Release Problem. *The 2nd International Conference on Software Engineering and Data Mining*, 166–171.

Jiao, H.-W., Wang, F.-H., & Chen, Y.-Q. (2014). An Effective Branch and Bound Algorithm for Minimax Linear Fractional Programming. *Journal of Applied Mathematics*, *2014*(1), 160262. https://doi.org/10.1155/2014/160262

Karabuk, S. (2009). A nested decomposition approach for solving the paratransit vehicle scheduling problem. *Transportation Research Part B:*

Methodological, *43*(4), 448–465. https://doi.org/10.1016/j.trb.2008.08.002

Khadilkar, H. (2022). Solving the capacitated vehicle routing problem with timing windows using rollouts and MAX-SAT. *2022 Eighth Indian Control Conference (ICC)*, 1–6. https://doi.org/10.1109/ICC56513.2022.10093678

Kovacs, A. A., Golden, B. L., Hartl, R. F., & Parragh, S. N. (2015). The Generalized Consistent Vehicle Routing Problem. *Transportation Science*, *49*(4), 796–816. https://doi.org/10.1287/trsc.2014.0529

Li, X., Liu, Y., Li, P., Zhao, Y., Dong, Y., & Jin, Y. (2024). *Green Vehicle Routing Problem with Capacitated Service Stations*. https://doi.org/10.21203/rs.3.rs-4495534/v1

Li, Y. (2018). *Deep Reinforcement Learning: An Overview* (arXiv:1701.07274). arXiv. http://arxiv.org/abs/1701.07274

Lin, B.-C., Liu, X.-F., & Mei, Y. (2022). Efficient Extended Ant Colony Optimization for Capacitated Electric Vehicle Routing. *2022 IEEE Symposium Series on Computational Intelligence (SSCI)*, 504–511. https://doi.org/10.1109/SSCI51031.2022.10022179

Luan, R. (2024). Logistics distribution route optimization of electric vehicles based on distributed intelligent system. *International Journal of Emerging Electric Power Systems*. https://doi.org/doi:10.1515/ijeeps-2023-0304

Ma, X., & Liu, C. (2024). Improved Ant Colony Algorithm for the Split Delivery Vehicle Routing Problem. *Applied Sciences*, *14*(12). https://doi.org/10.3390/app14125090

Mamoun, K. A., Hammadi, L., Novaes, A. G. N., Ballouti, A. E., & Cursi, E. S. D. (2022). An optimisation solution of Capacitated Vehicle Routing Problem (CVRP). *2022 11th International Symposium on Signal, Image, Video and Communications (ISIVC)*, 1–5. https://doi.org/10.1109/ISIVC54825.2022.9800726

Matijević, L., Ilin, V., Davidović, T., Jakšić-Krüger, T., & Pardalos, P. M. (2024). General VNS for asymmetric vehicle routing problem with time and capacity constraints. *Computers & Operations Research*, *167*, 106630. https://doi.org/10.1016/j.cor.2024.106630

Mavrovouniotis, M., Li, C., Ellinas, G., & Polycarpou, M. (2022). Solving the Electric Capacitated Vehicle Routing Problem with Cargo Weight. *2022 IEEE Congress on Evolutionary Computation (CEC)*, 1–8. https://doi.org/10.1109/CEC55065.2022.9870383

Mavrovouniotis, M., Menelaou, C., Timotheou, S., Ellinas, G., Panayiotou, C., & Polycarpou, M. (2020). A Benchmark Test Suite for the Electric Capacitated Vehicle Routing Problem. *2020 IEEE Congress on Evolutionary Computation (CEC)*, 1–8. https://doi.org/10.1109/CEC48606.2020.9185753

Mir Ehsan Hesam Sadati, V. A., & Çatay, B. (2022). Electric vehicle routing problem with flexible deliveries. *International Journal of Production*

Research, *60*(13), 4268–4294. https://doi.org/10.1080/00207543.2022.2032451

Mohammed, M. A., Ghani, M. K. A., Hamed, R. I., Mostafa, S. A., Ibrahim, D. A., Jameel, H. K., & Alallah, A. H. (2017). Solving vehicle routing problem by using improved K-nearest neighbor algorithm for best solution. *Journal of Computational Science*, *21*, 232–240. https://doi.org/10.1016/j.jocs.2017.04.012

Muazu, A. A., Hashim, A. S., & Sarlan, A. (2022). Review of Nature Inspired Metaheuristic Algorithm Selection for Combinatorial t-Way Testing. *IEEE Access*, *10*, 27404–27431. https://doi.org/10.1109/ACCESS.2022.3157400

Mukherjee, D., & Acharyya, S. (2011). New variants of Ant Colony Optimization for Network Routing. *14th International Conference on Computer and Information Technology (ICCIT 2011)*, 445–450. https://doi.org/10.1109/ICCITechn.2011.6164831

Nie, Z.-H., Yang, Q., Zhang, E., Liu, D., & Zhang, J. (2022). Ant Colony optimization for Electric Vehicle Routing Problem with Capacity and Charging Time Constraints. *2022 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, 480–485. https://doi.org/10.1109/SMC53654.2022.9945248

Phu-Ang, A. (2023). Solving the Capacitated Electric Vehicle (EV) Routing Problem by The Differential Evolutionary Algorithm with Adaptive K-Means. *2023 Joint International Conference on Digital Arts, Media and*

Technology with ECTI Northern Section Conference on Electrical, Electronics, Computer and Telecommunications Engineering (ECTI DAMT & NCON), 225–228. https://doi.org/10.1109/ECTIDAMTNCON57770.2023.10139512

Pujowidianto, N. A. (2017). Constrained optimization via simulation models for new product innovation. *IOP Conference Series: Materials Science and Engineering*, *273*(1), 012024. https://doi.org/10.1088/1757-899X/245/1/012024

Pustilnik, M., & Borrelli, F. (2024). *Clustering Heuristics for Robust Energy Capacitated Vehicle Routing Problem (ECVRP)* (arXiv:2403.13906). arXiv. http://arxiv.org/abs/2403.13906

Rand, G. K. (2024). From Operational Research Quarterly to Journal of the Operational Research Society: The early editors. *Journal of the Operational Research Society*, *75*(1), 3–12. https://doi.org/10.1080/01605682.2023.2244339

Rastani, S., & Çatay, B. (2023). A large neighborhood search-based matheuristic for the load-dependent electric vehicle routing problem with time windows. *Annals of Operations Research*, *324*(1), 761–793. https://doi.org/10.1007/s10479-021-04320-9

Rodrigues, S. B., & Yamashita, D. S. (2010). An exact algorithm for minimizing resource availability costs in project scheduling. *European Journal of Operational Research*, *206*(3), 562–568. https://doi.org/10.1016/j.ejor.2010.03.008

Sánchez, D. G., Tabares, A., Faria, L. T., Rivera, J. C., & Franco, J. F. (2022). A Clustering Approach for the Optimal Siting of Recharging Stations in the Electric Vehicle Routing Problem with Time Windows. *Energies*, *15*(7), 2372. https://doi.org/10.3390/en15072372

Shami, T. M., El-Saleh, A. A., Alswaitti, M., Al-Tashi, Q., Summakieh, M. A., & Mirjalili, S. (2022). Particle Swarm Optimization: A Comprehensive Survey. *IEEE Access*, *10*, 10031–10061. https://doi.org/10.1109/ACCESS.2022.3142859

Shukla, A., Pandey, H. M., & Mehrotra, D. (2015). Comparative review of selection techniques in genetic algorithm. *2015 International Conference on Futuristic Trends on Computational Analysis and Knowledge Management (ABLAZE)*, 515–519. https://doi.org/10.1109/ABLAZE.2015.7154916

Singh, V. P., Sharma, K., & Chakraborty, D. (2023). Solving capacitated vehicle routing problem with demands as fuzzy random variable. *Soft Computing*, *27*(21), 16019–16039. https://doi.org/10.1007/s00500-023-08888-1

Stamadianos, T., Kyriakakis, N. A., Marinaki, M., & Marinakis, Y. (2023). Routing Problems with Electric and Autonomous Vehicles: Review and Potential for Future Research. *Operations Research Forum*, *4*(2), 46. https://doi.org/10.1007/s43069-023-00228-1

Su, Z., & Chen, P. (2023). *Driver-Centric Capacitated Electric Vehicle Routing Problem: A Human Energy-Aware Approach*. https://doi.org/10.2139/ssrn.4584804

Tablada, T., Davis, C., & Palumbo, N. (2024). An Ant Colony Algorithm Approach to the Vehicle Routing Problem and Variants. *2024 Systems and Information Engineering Design Symposium (SIEDS)*, 51–56. https://doi.org/10.1109/SIEDS61124.2024.10534666

Tahami, H., Rabadi, G., & Haouari, M. (2020). Exact approaches for routing capacitated electric vehicles. *Transportation Research Part E: Logistics and Transportation Review*, *144*, 102126. https://doi.org/10.1016/j.tre.2020.102126

Tang, M., Li, B., Liu, H., Zhuang, W., Li, Z., & Peng, J. (2022). Energy-oriented Routing Strategy of Electric Vehicle: An End-to-End Reinforcement Learning Approach. *2022 6th CAA International Conference on Vehicular Control and Intelligence (CVCI)*, 1–7. https://doi.org/10.1109/CVCI56766.2022.9964505

Ting, T. O., Yang, X.-S., Cheng, S., & Huang, K. (2015). Hybrid Metaheuristic Algorithms: Past, Present, and Future. In X.-S. Yang (Ed.), *Recent Advances in Swarm Intelligence and Evolutionary Computation* (pp. 71–83). Springer International Publishing. https://doi.org/10.1007/978-3-319-13826-8_4

Triki, E., Collette, Y., & Siarry, P. (2005). A theoretical study on the behavior of simulated annealing leading to a new cooling schedule. *European*

Journal of Operational Research, *166*(1), 77–92. https://doi.org/10.1016/j.ejor.2004.03.035

Udomsakdigool, A., & Kachitvichyanukul, V. (2008). Multiple-Colony Ant Algorithm with Forward–Backward Scheduling Approach for Job-Shop Scheduling Problem. In A. H. S. Chan & S.-I. Ao (Eds.), *Advances in Industrial Engineering and Operations Research* (Vol. 5, pp. 39–55). Springer US. https://doi.org/10.1007/978-0-387-74905-1_4

Valadarsky, A., Schapira, M., Shahaf, D., & Tamar, A. (2017). *A Machine Learning Approach to Routing* (arXiv:1708.03074). arXiv. http://arxiv.org/abs/1708.03074

Volna, E. (2016). Genetic algorithms for the vehicle routing problem. *AIP Conference Proceedings*, *1738*(1), 120002. https://doi.org/10.1063/1.4951885

Vu, X.-H., Silaghi, M.-C., Sam-Haroud, D., & Faltings, B. (2007). *Branch-and-Prune Search Strategies for Numerical Constraint Solving* (arXiv:cs/0512045). arXiv. http://arxiv.org/abs/cs/0512045

Wang, Z., Yan, Q., Wang, L., & Hei, X. (2023). Solving the Vehicle Routing Problem with Time Window based on Reinforcement Learning. *2023 19th International Conference on Natural Computation, Fuzzy Systems and Knowledge Discovery (ICNC-FSKD)*, 1–7. https://doi.org/10.1109/ICNC-FSKD59587.2023.10280869

Xiang, X., Qiu, J., Xiao, J., & Zhang, X. (2020). Demand coverage diversity based ant colony optimization for dynamic vehicle routing problems.

Engineering Applications of Artificial Intelligence, *91*, 103582. https://doi.org/10.1016/j.engappai.2020.103582

Yavşan, E., & İlhan, İ. (2022). An Ant Colony Optimization Based Real-time Mobile Application for the Capacitated Vehicle Routing Problem. *Hittite Journal of Science and Engineering*, *9*(4), 263–273. https://doi.org/10.17350/HJSE19030000279

Zahedi, F., Kia, H., & Khalilzadeh, M. (2023). A hybrid metaheuristic approach for solving a bi-objective capacitated electric vehicle routing problem with time windows and partial recharging. *Journal of Advances in Management Research*, *20*(4), 695–729. https://doi.org/10.1108/JAMR-01-2023-0007

Zhang, S., Gajpal, Y., Appadoo, S. S., & Abdulkader, M. M. S. (2018). Electric vehicle routing problem with recharging stations for minimizing energy consumption. *International Journal of Production Economics*, *203*, 404–413. https://doi.org/10.1016/j.ijpe.2018.07.016

Zhang, Y., Li, M., Chen, Y., Chiang, Y.-Y., & Hua, Y. (2023). A Constraint-Based Routing and Charging Methodology for Battery Electric Vehicles With Deep Reinforcement Learning. *IEEE Transactions on Smart Grid*, *14*(3), 2446–2459. https://doi.org/10.1109/TSG.2022.3214680

Zhang, Z., & Zou, K. (2017). Simple ant colony algorithm for combinatorial optimization problems. *2017 36th Chinese Control Conference (CCC)*, 9835–9840. https://doi.org/10.23919/ChiCC.2017.8028925

Zhao, M. (2023). Application of an Improved Ant Colony Algorithm in Vehicle Routing Problem. *2023 4th International Conference on Computer Engineering and Intelligent Control (ICCEIC)*, 270–273. https://doi.org/10.1109/ICCEIC60201.2023.10426644

Zhou, S., Wei, J., & Yan, C.-B. (2022). A Multi-stage Algorithm for the Capacitated Vehicle Routing Problem with Two-dimensional Loading and Time Windows. *2022 IEEE 18th International Conference on Automation Science and Engineering (CASE)*, 1714–1719. https://doi.org/10.1109/CASE49997.2022.9926490

APPENDICES

APPENDIX A

Python Code for Input Data

```
import pandas as pd
import numpy as np

# Example demand data
demand_data = {
    'Node_ID': [
        'Depot', 'Customer_1', 'Customer_2', 'Customer_3', 'Customer_4',
        'Customer_5', 'Customer_6', 'Customer_7', 'Customer_8',
        'Customer_9', 'Customer_10', 'Charging_Station'
    ],
    'Demand': [0, 10, 20, 15, 25, 5, 10, 20, 15, 30, 15, 0]
}

# Create DataFrame and save to CSV
demand_df = pd.DataFrame(demand_data)
demand_df.to_csv('demand.csv', index=False)

# Example distance matrix (12x12 matrix)
distance_matrix = np.array([
    [0, 5, 10, 7, 8, 6, 9, 4, 3, 10, 8, 5],
    [5, 0, 3, 6, 7, 8, 4, 9, 10, 3, 4, 6],
    [10, 3, 0, 2, 5, 9, 7, 4, 3, 8, 6, 7],
    [7, 6, 2, 0, 4, 8, 10, 3, 2, 7, 9, 6],
    [8, 7, 5, 4, 0, 3, 6, 7, 8, 2, 5, 7],
    [6, 8, 9, 8, 3, 0, 2, 5, 6, 4, 3, 5],
    [9, 4, 7, 10, 6, 2, 0, 3, 5, 8, 7, 9],
    [4, 9, 4, 3, 7, 5, 3, 0, 2, 6, 9, 3],
    [3, 10, 3, 2, 8, 6, 5, 2, 0, 7, 4, 4],
    [10, 3, 8, 7, 2, 4, 8, 6, 7, 0, 3, 8],
    [8, 4, 6, 9, 5, 3, 7, 9, 4, 3, 0, 2],
    [5, 6, 7, 6, 7, 5, 9, 3, 4, 8, 2, 0]
])

# Create DataFrame for distance matrix
node_names = [
    'Depot', 'Customer_1', 'Customer_2', 'Customer_3', 'Customer_4',
    'Customer_5', 'Customer_6', 'Customer_7', 'Customer_8',
    'Customer_9', 'Customer_10', 'Charging_Station'
]
```

```
distance_matrix_df = pd.DataFrame(distance_matrix, columns=node_names, index=node_names)

# Save to CSV
distance_matrix_df.to_csv('distance_matrix.csv')

print("CSV files 'demand.csv' and 'distance_matrix.csv' have been created.")
```

APPENDIX B

Python Code for ACO Algorithm

```
import numpy as np
import pandas as pd

# Set a random seed for reproducibility
np.random.seed(42)

# Load demand data
demand_data = pd.read_csv('demand.csv')
demands = demand_data['Demand'].values

# Load distance matrix data
distance_matrix = pd.read_csv('distance_matrix.csv', index_col=0).values

# Parameters for ACO
num_customers = 10  # Number of customers to visit
max_iterations = 200  # Number of iterations
battery_capacity = 150  # Maximum battery capacity of the EV
vehicle_capacity = 100  # Maximum load capacity of the EV
start_node = 0  # Depot is node 0
num_ants = 50  # Number of ants
alpha = 5  # Influence of pheromone on direction
```

```
beta = 5  # Influence of heuristic value (distance)
rho = 0.1  # Evaporation rate
Q = 10  # Pheromone deposit factor

start_time = time.time()

# Initialize pheromone levels: Initially, all edges have the same pheromone level
num_nodes = len(demands)
pheromone_matrix = np.ones((num_nodes, num_nodes)) / num_nodes

def print_pheromone_matrix(pheromone_matrix):
    print("Pheromone Matrix:")
    print(pheromone_matrix)
    print_pheromone_matrix(pheromone_matrix)
# Validate the distance matrix against battery capacity
if np.any(distance_matrix > battery_capacity):
    raise ValueError("Invalid battery capacity input: Some distances exceed the EV's maximum battery capacity.")

#-----------------------------------------------------------------------------------

#Step 2: Deploay ants to construct solutions
# Function to calculate the probability of moving to the next node
def calculate_transition_probabilities(current_node, visited_nodes, pheromone_matrix, distance_matrix, alpha, beta, start_node):
```

```
    num_nodes = len(pheromone_matrix)
    probabilities = np.zeros(num_nodes)

    for next_node in range(num_nodes):
        # Exclude already visited nodes and prevent consecutive visits to depot or charging station
        if next_node not in visited_nodes and \
           (next_node != start_node or current_node != start_node) and \
           (next_node != num_nodes - 1 or current_node != num_nodes - 1):
            tau = pheromone_matrix[current_node][next_node] ** alpha
            eta = (1 / distance_matrix[current_node][next_node]) ** beta
            probabilities[next_node] = tau * eta

    # Normalize the probabilities
    probabilities_sum = np.sum(probabilities)
    if probabilities_sum > 0:
        probabilities /= probabilities_sum
    return probabilities

# Function to initialize the pheromone levels and deploy ants to build routes
def initialize_pheromones_and_build_routes(num_ants, start_node, pheromone_matrix, distance_matrix, demands, battery_capacity, vehicle_capacity, alpha, beta):
    solutions = []
    for ant in range(num_ants):
```

```
        '''print(f"\nAnt {ant + 1} is constructing a route:")'''
        solution = construct_feasible_route(start_node, pheromone_matrix, distance_matrix, demands, battery_capacity, vehicle_capacity, alpha, beta)
        solutions.append(solution)
    return solutions


# Function to construct a feasible route considering battery and load constraints
def construct_feasible_route(start_node, pheromone_matrix, distance_matrix, demands, battery_capacity, vehicle_capacity, alpha, beta):
    num_nodes = len(pheromone_matrix)
    charging_station_node = num_nodes - 1  # The last node is always the charging station

    visited_nodes = [start_node]
    current_node = start_node
    battery_level = battery_capacity
    load = 0  # Current load of the EV

    unvisited_customers = set(range(1, num_nodes - 1))  # All customers except depot and charging station

    while unvisited_customers:
        probabilities = calculate_transition_probabilities(current_node, visited_nodes, pheromone_matrix, distance_matrix, alpha, beta, start_node)
        next_node = np.random.choice(range(num_nodes), p=probabilities)
```

if next_node == charging_station_node or next_node not in unvisited_customers:

continue # Skip the charging station initially and avoid revisiting customers

demand = demands[next_node]

distance_to_next = distance_matrix[current_node][next_node]

distance_to_charging_station = distance_matrix[current_node][charging_station_node]

distance_to_depot = distance_matrix[current_node][start_node]

Check if the EV can serve the next customer without exceeding its load or battery capacity

if (load + demand <= vehicle_capacity) and (battery_level >= distance_to_next):

Check if the EV can still reach the charging station or depot afterward

if battery_level - distance_to_next >= min(distance_to_charging_station, distance_to_depot):

visited_nodes.append(next_node)

unvisited_customers.remove(next_node)

battery_level -= distance_to_next

load += demand

current_node = next_node

```
            else:
                # If not enough battery to reach the next customer and then the depot/charging station, go to charging station first
                visited_nodes.append(charging_station_node)
                battery_level = battery_capacity  # Recharge fully
                current_node = charging_station_node

        else:
            # If the load or battery cannot support going to the next customer, return to the depot or go to the charging station
            if battery_level >= distance_to_depot:
                visited_nodes.append(start_node)
                battery_level = battery_capacity  # Recharge at the depot
                load = 0  # Unload at the depot
                current_node = start_node
            elif battery_level >= distance_to_charging_station:
                visited_nodes.append(charging_station_node)
                battery_level = battery_capacity  # Recharge fully
                current_node = charging_station_node
            else:
                break  # Exit and return the current partial route

    # After visiting all customers, check if EV can return to the depot
    if not unvisited_customers:
        distance_to_depot = distance_matrix[current_node][start_node]
```

```
        if battery_level >= distance_to_depot:
            battery_level -= distance_to_depot
            visited_nodes.append(start_node)
        else:
            # If not enough battery to return to the depot, visit charging station before returning
            visited_nodes.append(charging_station_node)
            battery_level = battery_capacity  # Recharge fully
            visited_nodes.append(start_node)

    return visited_nodes



# Deploy ants to construct solutions with full customer visits and return to depot
def deploy_ants(num_ants, start_node, pheromone_matrix, distance_matrix, demands, battery_capacity, vehicle_capacity, alpha, beta):
    solutions = []
    num_nodes = len(pheromone_matrix)
    for i in range(num_ants):
        solution = construct_feasible_route(start_node, pheromone_matrix, distance_matrix, demands, battery_capacity, vehicle_capacity, alpha, beta)

        # Verify that all customers were visited
        if len(set(solution)) != num_nodes:
        else:
```

```
        solutions.append(solution)
    return solutions

#-----------------------------------------------------------------------------------

#Step 3: Each ant build a route
solutions = initialize_pheromones_and_build_routes(num_ants, start_node, pheromone_matrix, distance_matrix, demands, battery_capacity, vehicle_capacity, alpha, beta)

#-----------------------------------------------------------------------------------

#Step 4: Calculate total distance
# Function to calculate the total distance of a given route
def calculate_total_distance(route, distance_matrix):
    total_distance = 0
    for i in range(len(route) - 1):
        total_distance += distance_matrix[route[i]][route[i + 1]]
    return total_distance

# Calculate the total distance for each route (step 4)
distances = []
for i, solution in enumerate(solutions):
    total_distance = calculate_total_distance(solution, distance_matrix)
```

```
    distances.append(total_distance)
    solution_str = [int(node) for node in solution]  # Convert np.int64 to int
    print(f"Ant {i + 1} Route: {solution_str}\nTotal Distance: {total_distance}")

#-------------------------------------------------------------------------------------

#Step 5: Identify best route
# Function to identify the best route based on the criteria
def identify_best_route(solutions, distances, num_customers):
    best_distance = float('inf')
    best_route = None

    for i, route in enumerate(solutions):
        if len(set(route)) == num_customers + 2 and route[-1] == 0:  # Check if all
customers are visited and return to depot
            if distances[i] < best_distance:
                best_distance = distances[i]
                best_route = route

    return best_route, best_distance

best_route, best_distance = identify_best_route(solutions, distances,
num_customers)

#-------------------------------------------------------------------------------------
```

```
#Step 6: Update pheromone levels
# Function to update pheromone levels based on the routes and distances
def update_pheromone_levels(pheromone_matrix, solutions, distances, rho, Q):
    num_nodes = len(pheromone_matrix)

    # Evaporate pheromone levels
    pheromone_matrix *= (1 - rho)

    # Update pheromones based on solutions
    for i, route in enumerate(solutions):
        contribution = Q / distances[i]
        for j in range(len(route) - 1):
            pheromone_matrix[route[j]][route[j + 1]] += contribution
            pheromone_matrix[route[j + 1]][route[j]] += contribution  # Since the graph is undirected

    return pheromone_matrix

pheromone_matrix = update_pheromone_levels(pheromone_matrix, solutions, distances, rho, Q)

#-------------------------------------------------------------------------------------

#Step 7: Apply local search
```

```
# Function to perform 2-opt swap
def two_opt_swap(route, i, k):
    new_route = route[:i] + route[i:k + 1][::-1] + route[k + 1:]
    return new_route

# Function to apply local search (2-opt)
def apply_local_search(route, distance_matrix):
    best_route = route
    best_distance = calculate_total_distance(route, distance_matrix)

    improvement = True
    while improvement:
        improvement = False
        for i in range(1, len(route) - 2):
            for k in range(i + 1, len(route) - 1):
                new_route = two_opt_swap(best_route, i, k)
                new_distance = calculate_total_distance(new_route, distance_matrix)
                if new_distance < best_distance:
                    best_route = new_route
                    best_distance = new_distance
                    improvement = True

    return best_route, best_distance

end_time = time.time()
```

```
# Output the best route and its total distance
if best_route is not None:
    best_route_str = [int(node) for node in best_route]  # Convert np.int64 to int
    print(f"\nBest Route: {best_route_str}\nTotal Distance: {best_distance}")
else:
    print("No valid route found that visits all customers and returns to the depot.")



# Calculate and print computational time
computational_time = end_time - start_time
print(f"Computational Time: {computational_time:.4f} seconds")
```

APPENDIX C

Python Code Output

Ant 1 Route: [0, 10, 8, 5, 0, 7, 1, 0, 6, 2, 3, 11, 9, 0, 4, 0]

Total Distance: 595

Ant 2 Route: [0, 7, 5, 3, 2, 0, 10, 8, 9, 0, 6, 1, 11, 4, 0]

Total Distance: 402

Ant 3 Route: [0, 10, 7, 5, 3, 0, 6, 2, 9, 8]

Total Distance: 207

Ant 4 Route: [0, 10, 7, 5, 3, 0, 6, 2, 9, 8]

Total Distance: 207

Ant 5 Route: [0, 10, 7, 5, 3, 0, 6, 2, 9, 4, 0, 8, 11, 1, 0]

Total Distance: 550

Ant 6 Route: [0, 10, 7, 5, 0, 6, 2, 3, 0, 4, 9, 8]

Total Distance: 314

Ant 7 Route: [0, 10, 8, 5, 0, 7, 1, 0, 6, 2, 3, 11, 4, 0, 9, 11, 0]

Total Distance: 630

Ant 8 Route: [0, 10, 7, 5, 9, 0, 6, 2, 3, 0, 4, 1, 11, 0, 8, 11, 0]

Total Distance: 720

Ant 9 Route: [0, 6, 2, 3, 5, 0, 10, 7, 1, 0, 8, 11, 4, 9, 0]

Total Distance: 558

Ant 10 Route: [0, 10, 7, 5, 9, 0, 6, 2, 3, 0, 4, 1, 11, 0, 8, 11, 0]

Total Distance: 720

Ant 11 Route: [0, 7, 5, 3, 2, 0, 10, 8, 9, 0, 6, 1, 11, 4, 0]

Total Distance: 402

Ant 12 Route: [0, 10, 8, 5, 0, 7, 1, 0, 6, 2, 3, 11, 4, 0, 9, 11, 0]

Total Distance: 630

Ant 13 Route: [0, 10, 8, 5, 0, 6, 2, 3, 0, 7, 1, 0, 4, 9, 11, 0]

Total Distance: 542

Ant 14 Route: [0, 10, 7, 5, 9, 0, 6, 2, 3, 0, 4, 1, 11, 0, 8, 11, 0]

Total Distance: 720

Ant 15 Route: [0, 7, 10, 8, 0, 6, 2, 3, 5, 0, 4, 9, 11, 0, 1, 11, 0]

Total Distance: 588

Ant 16 Route: [0, 7, 5, 3, 2, 0, 10, 8, 9, 0, 6, 1, 11, 4, 0]

Total Distance: 402

Ant 17 Route: [0, 10, 7, 5, 3, 0, 6, 2, 9, 8]

Total Distance: 207

Ant 18 Route: [0, 7, 5, 3, 2, 0, 10, 8, 9, 0, 6, 1, 11, 4, 0]

Total Distance: 402

Ant 19 Route: [0, 10, 7, 5, 3, 0, 6, 2, 9, 8]

Total Distance: 207

Ant 20 Route: [0, 7, 5, 3, 2, 0, 10, 8, 9, 0, 6, 1, 11, 4, 0]

Total Distance: 402

Ant 21 Route: [0, 7, 5, 3, 2, 0, 10, 8, 9, 0, 6, 1, 11, 4, 0]

Total Distance: 402

Ant 22 Route: [0, 10, 7, 5, 3, 0, 6, 2, 9, 4, 0, 1, 11, 8, 11, 0]

Total Distance: 603

Ant 23 Route: [0, 10, 7, 5, 3, 0, 6, 2, 9, 8]

Total Distance: 207

Ant 24 Route: [0, 10, 7, 5, 3, 0, 6, 2, 9, 8]

Total Distance: 207

Ant 25 Route: [0, 10, 7, 5, 3, 0, 6, 2, 9, 4, 0, 1, 11, 8, 11, 0]

Total Distance: 603

Ant 26 Route: [0, 10, 7, 5, 3, 0, 6, 2, 9, 4, 0, 8, 11, 1, 0]

Total Distance: 550

Ant 27 Route: [0, 10, 7, 5, 3, 0, 6, 2, 9, 8]

Total Distance: 207

Ant 28 Route: [0, 2, 3, 5, 7, 0, 10, 8, 9, 0, 6, 4, 0, 1, 11, 0]

Total Distance: 542

Ant 29 Route: [0, 10, 7, 5, 3, 0, 6, 2, 9, 8]

Total Distance: 207

Ant 30 Route: [0, 10, 7, 5, 3, 0, 6, 2, 9, 4, 0, 1, 11, 8, 11, 0]

Total Distance: 603

Ant 31 Route: [0, 10, 7, 5, 0, 6, 2, 3, 0, 4, 9, 8]

Total Distance: 314

Ant 32 Route: [0, 10, 7, 5, 0, 6, 2, 3, 0, 4, 9, 8]

Total Distance: 314

Ant 33 Route: [0, 7, 5, 3, 2, 0, 10, 8, 9, 0, 6, 1, 11, 4, 0]

Total Distance: 402

Ant 34 Route: [0, 10, 7, 5, 3, 0, 6, 2, 9, 8]

Total Distance: 207

Ant 35 Route: [0, 6, 2, 3, 5, 0, 10, 7, 1, 0, 4, 9, 8, 11, 0]

Total Distance: 481

Ant 36 Route: [0, 10, 8, 5, 0, 6, 2, 3, 0, 7, 9, 11, 4, 0, 1, 11, 0]

Total Distance: 623

Ant 37 Route: [0, 7, 5, 3, 2, 0, 10, 8, 9, 0, 6, 1, 11, 4, 0]

Total Distance: 402

Ant 38 Route: [0, 10, 7, 5, 3, 0, 6, 2, 9, 4, 0, 1, 11, 8, 11, 0]

Total Distance: 603

Ant 39 Route: [0, 10, 7, 5, 3, 0, 6, 2, 9, 8]

Total Distance: 207

Ant 40 Route: [0, 10, 7, 5, 3, 0, 6, 2, 9, 8]

Total Distance: 207

Ant 41 Route: [0, 10, 8, 5, 0, 7, 1, 0, 6, 2, 3, 11, 4, 0, 9, 11, 0]

Total Distance: 630

Ant 42 Route: [0, 10, 7, 5, 0, 6, 2, 3, 0, 9, 11, 1, 0, 4, 11, 8, 11, 0]

Total Distance: 773

Ant 43 Route: [0, 6, 2, 3, 5, 0, 10, 7, 1, 0, 4, 9, 8, 11, 0]

Total Distance: 481

Ant 44 Route: [0, 7, 5, 3, 2, 0, 10, 8, 9, 0, 6, 1, 11, 4, 0]

Total Distance: 402

Ant 45 Route: [0, 7, 5, 8, 0, 10, 4, 9, 2, 0, 6, 1, 11, 3, 11, 0]

Total Distance: 596

Ant 46 Route: [0, 7, 5, 3, 2, 0, 10, 8, 9, 0, 6, 4, 0, 1, 11, 0]

Total Distance: 542

Ant 47 Route: [0, 10, 7, 5, 3, 0, 6, 2, 9, 4, 0, 1, 11, 8, 11, 0]

Total Distance: 603

Ant 48 Route: [0, 10, 7, 5, 3, 0, 6, 1, 9, 11, 0, 2, 4, 11, 8, 11, 0]

Total Distance: 684

Ant 49 Route: [0, 7, 10, 8, 0, 6, 5, 3, 2, 0, 9, 11, 1, 0, 4, 0]

Total Distance: 598

Ant 50 Route: [0, 10, 7, 5, 3, 0, 6, 2, 9, 8]

Total Distance: 207

Best Route: [0, 7, 5, 3, 2, 0, 10, 8, 9, 0, 6, 1, 11, 4, 0]

Total Distance: 402

turnitin checking

ORIGINALITY REPORT

13%	8%	10%	4%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	eprints.utar.edu.my Internet Source	1%
2	Submitted to Universiti Tunku Abdul Rahman Student Paper	<1%
3	arxiv.org Internet Source	<1%
4	Submitted to Johns Hopkins Unversity Student Paper	<1%
5	www.mdpi.com Internet Source	<1%
6	Prasanna Balaprakash. "Improvement Strategies for the F-Race Algorithm: Sampling Design and Iterative Refinement", Lecture Notes in Computer Science, 2007 Publication	<1%
7	ebin.pub Internet Source	<1%
8	Bo-Cheng Lin, Xiao-Fang Liu, Yi Mei. "Efficient Extended Ant Colony Optimization for	<1%

Universiti Tunku Abdul Rahman			
Form Title : Supervisor's Comments on Originality Report Generated by Turnitin for Submission of Final Year Project Report (for Undergraduate Programmes)			
Form Number: FM-IAD-005	Rev No.: 0	Effective Date: 01/10/2013	Page No.: 126 of 1

FACULTY OF SCIENCE

Full Name(s) of Candidate(s)	Samuel Tiong Fu Wei
ID Number(s)	20ADB04885
Programme / Course	BACHELOR OF SCIENCE (HONOURS) STATISTICAL COMPUTING AND OPERATIONS RESEARCH
Title of Final Year Project	Optimizing Capacitated Electric Vehicle Route In Logistic Operations

Similarity	**Supervisor's Comments (Compulsory if parameters of originality exceeds the limits approved by UTAR)**
Overall similarity index: __13__ % **Similarity by source** Internet Sources: ___8___ % Publications: ___10___ % Student Papers: ___4___ %	**nil**
Number of individual sources listed of more than 3% similarity: ___0____	
Parameters of originality required and limits approved by UTAR are as follows: **(i) Overall similarity index is 20% and below , and** **(ii) Matching of individual sources listed must be less than 3% each , and** **(iii) Matching texts in continuous block must not exceed 8 words** ***Note: Parameters (i) – (ii) shall exclude quotes, bibliography and text matches which are less than 8 words.***	

Note Supervisor/Candidate(s) is/are required to provide softcopy of full set of the originality report to Faculty/Institute

Based on the above results, I hereby declare that I am satisfied with the originality of the Final Year Project Report submitted by my student(s) as named above.

______________________________ Signature of Supervisor

Name: NUR INTAN LIYANA BINTI MOHD AZMI

Date: 02/09/2024

______________________________ Signature of Co-Supervisor

Name: ______________________

Date: ______________________