Vehicle Routing

Project Report

Submitted by

Revati Satyawan Sawant, ED19B025

in partial fulfillment of requirements
for the award of the dual degree of

BACHELOR OF TECHNOLOGY in ENGINEERING DESIGN

and

MASTER OF TECHNOLOGY in DATA SCIENCE

Guided by

Prof. G. Srinivasan

Department of Engineering Design
Indian Institute of Technology Madras
Chennai - 600036, India

May 2024

Acknowledgements

I would like to express my sincere gratitude to my advisor, Prof. G. Srinivasan, for their unwavering support, guidance, and mentorship throughout my project journey. Without their expertise, patience, and encouragement, I would not have been able to progress this far.

I am particularly grateful for his flexibility and understanding, which allowed me to manage my time and stress levels more effectively during this process. Their support also allowed me to sit for job placements and pursue other opportunities while working on my project.

I would like to express my sincere thanks to Prof. Rahul Marathe, who provided valuable feedback during the reviews. I received incredible clarity from our conversations.

I am also thankful to my colleagues and friends who have provided invaluable support and assistance during the course of my research. In particular, I would like to thank the batch mates of the data science program: Sahil Girhepuje for the helpful suggestions and for being a constant source of support and motivation.

I am grateful to my family and loved ones for their love and support. Their belief in me and encouragement have been a constant source of strength and inspiration throughout this journey.

Thank you all for your invaluable contributions to my project.

Abstract

Employee pick up problem addresses the transportation of employees from their respective places to factory or an office. This is a challenging problem for every organisation particularly when the number of people employed is large. Today, companies prefer to outsource these operations to service providers.

These types of problems come under the category of vehicle routing problems (VRP) with line hauls and backhauls. In this report we address the employee pick up problem under different situations. Since the service is outsourced, the buses may or may not start every trip from the factory but could be parked in depots when idle and start from these depots to pick up employees for the next shift. Sometimes due to various factors the number of employees to be picked up might change leading to stochasticity. This results in interesting variants of the conventional vehicle routing problem.

The number of employees and their boarding points would change frequently which necessitates rerouting of the buses. From the service provider perspective, the objective is to provide minimum cost (distance) service. However the rerouting could result in increased travel time for some of the employees. We consider the objective of minimising the total travel distance by the vehicles and understand the role of stochasticity to formulate a relevant problem. Several models and their performances are documented in the following report with an intent to implement this on real data from a leading automotive manufacturer. These formulations are tested with randomly generated test problems, since there are no existing benchmark problems with known results for our objective. Objectives such as minimising total distance and minimising regret are considered in this thesis.

These methods are further tested to solve a real-life problem from a leading automobile manufacturing company in Chennai involving picking up of about 3000 employees.

Contents

List of Figures

Chapter 1

Introduction

1.1 Motivation

Providing transportation for their employees is a common industry practice these days particularly when the company or a factory is located outside the city. Companies working three shifts or working 24 x 7 provide this facility to their employees. At times, concern for safety of employees and government regulations also necessitate that companies provide transportation for their employees. Employees working in cyclic shifts and women joining the workforce increase the need for safe and reliable transportation. Providing transportation also helps in reducing carbon emission and hence the environmental impact. Eliminating a car from road would reduce about 4.6 tonnes (United States Environmental Protection Agency) of carbon dioxide per a year. Companies can also develop their corporate reputation by making themselves environment friendly. To become an employer of choice, companies offer more benefits in addition to the salary. Additionally companies can also increase the retention rate of employees by providing transportation. This is also considered as a way for expanding the employees working hours. On an average employee spends more time for commuting if they travel by their own means. This also affects the productivity of the employees. All these factors trigger the necessity for the companies to provide transportation for their employees.

The company can either provide its own transportation systems or hire it from a third-party logistics service provider. Transportation spending is generally observed to be within top five facility service expenditures for companies. Owning the vehicles would increase the financial burden for the companies especially when the number of employees is high. Increasingly these transportation activities are outsourced to third party logistics service providers. These third-party service providers will do the job of aggregators and provide on time reliable service to their employees. There are many challenges in the management of transportation for both the company and third-party service provider. Minimising the transportation cost is one of the key objectives that the company looks at while the employee looks at satisfaction that comes from travelling the minimum distance. There needs to be a good trade off between the transportation cost and employee satisfaction. Flexibility, cost and time are the important factors that an employee would consider for the mode of transportation. The employee would not prefer transportation provided by the company when the above factors are not considered. Increase in the number of employees and attrition will change the number of boarding points as well as the number of people boarding the buses frequently. This becomes a challenge to the service provider and rerouting has to be done often. This would cause some employees to spend more time in the vehicle and sometimes can lead to a route with increased travel times. In most industries, these transport processes are planned by mind mapping the bus stops they can club together and they don't have a well thought out metric to measure the employee satisfaction. Optimization of the routes considering all the above factors would be an important research problem for organisations.

1.2 Conventional Vehicle Routing Problem

Vehicle routing problem (VRP) is commonly used to address these types of problems. The most fundamental routing problem is the Travelling Salesman Problem (TSP) where the salesman visits a set of cities once and only once and comes back to the starting point travelling the minimum distance. When we add capacity constraints to the TSP problem it becomes a VRP problem. It was first introduced by Dantzig and Ramser (1959) as a Truck dispatching problem for optimum routing of gasoline delivery trucks between a bulk terminal and service stations supplied by the terminal. The problem has attracted the attention of both theory and practice due to its extensive applications. The objective of this conventional vehicle routing problem is to minimise the total distance travelled by the fleets in such a way that:

1. Every vertex (node) is visited exactly once by a vehicle
2. All vehicles' routes start and ends at factory (depot)
3. Quantity allocated to each vehicle should not exceed the vehicle capacity

The employee pick up and drop problem can be modelled as a VRP. Each bus stop corresponds to a vertex (node) and the demand associated with it is the number of employees to be picked from or dropped off at that node. Employees are to be picked up from different places in the city and brought to the plant. At the end of the shift they will be dropped from where they were picked. The length of each route cannot exceed a predetermined distance limit. This ensures that employees do not travel excess distance or have to start off very early from their place. Every vehicle will have capacity and will be assigned to the subset of the bus stops according to the distance limit and the bus capacity.

Minimising the total distance travelled by all fleet would be an ideal objective from the point of view of the organisation but may not be ideal from the employee point of view. For instance, an employee who can reach the office in less time directly may have to travel more distance in a bus when demand from different nodes are combined to meet the capacity of the bus. Increasing the number of fleet would solve this problem, but the fixed cost for every additional fleet is high and the organisation may not be willing to consider this option always. Companies would want to optimise the route with the available number of fleet. Also conventional VRP does not solve the problem considering stochasticity. When it comes to solving large real life problems optimal solutions by conventional VRP formulations are complicated and there is a need for heuristic solutions.

1.3 Objective

In our research we present a variant of the VRP problem motivated by a real-life problem from a leading automobile company which involves picking up employees with one or many vehicles and multiple depots for parking of the vehicles. Stochasticity at nodes is also applied and tested. We use a term called Depot to denote a place where a vehicle is parked after serving a preceding trip and starts from the same place to pick up employees for the subsequent trip. We also assume that there is a linear relationship between the travel time and distance. We have two objectives,

- To minimise the total distance travelled by all vehicles used in this operation
- To minimise the regret

The number of employees to be picked up and dropped would change frequently due to attrition and new recruits resulting in rerouting of the vehicles and stochasticity at a node. The inconvenience caused due to this is being factored into regret.

This employee pick up problem is modelled as an integer programming problem considering both the objectives. These formulations are tested with the randomly generated test problems, since there are no existing benchmark problems to compare. We also develop meta heuristics algorithms based on Tabu search and Genetic Algorithm to solve large sized problems. We also model a few stochastic models to handle the volatility at each node. We solve a real-life problem from a leading automobile company in Chennai involving picking and dropping of 3000 employees This report is organised as follows.

- A brief literature review related to the problem statement is introduced in chapter 2.

- In Chapter 3 we formulate and solve mixed integer linear programming formulations considering the objective of minimising total distance. We also develop GENETIC and TABU Search algorithms. We compare the solutions found using all the models.

- In Chapter 4 we introduce the concept of stochastic VRP, develop models and apply it to a smaller problem statement. We also explain the extensions and implications of stochasticity to our original employee pick up problem

- In Chapter 5 we summarise the report and provides a wholistic perspective of everything achieved so far.

- In Chapter 6 we discuss the future scope for the report.

Chapter 2

Literature Review

2.1 Travelling salesman problem (TSP)

The Travelling salesman problem is one of the oldest researched problems and it roots are ambiguous. The basic problem is that a travelling salesman who has to visit prespecified number of cities needs to know what route should he take to cover all the places travelling minimum distance and returns back to the place where he started by visiting all the cities once and only once. Several practical and theoretical problems are related to the TSP and the problem is well researched for the past sixty years. One of the first academic research reported is by Dantzig et al. (1954)

The travelling salesman problem has its roots in graph theory. A simple graph G = (V, E) which has set of vertices V, is the set of cities needs to be visited including the starting city and edges E connect the vertices V. Sometimes edges won't be available between two vertices owing to practical situations like road blocks, road unavailability etc. So the tour is a certain sequence of the subset of the edges. A tour is a Hamiltonian cycle that is each vertex has to be visited only once and it should end at the starting place. Furthermore, there would a weight associated with each edge. These weights can be cost, distance or time required to travel from one vertex to the other vertex and can be constructed in a matrix form. This matrix usually satisfies the triangle inequality. This states that given three vertices and the distance among them, sum of any two distances is greater than (or equal to) the third. In more practical terms it is less expensive to visit directly than going through one or more cities. The distance between the vertices is usually the Euclidean distance. For instance vertex Vi and Vj can be located in a graph by $V_i = (x_1\ , y_1\ )$ and $V_j = (x_2\ , y_2\ )$. The following formula is used to calculate the Euclidean distance.

$d(\vec{V_1}, \vec{V_2}) = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$

Euclidean distance also satisfies the triangle inequality. This matrix can be symmetric when distance between the two vertices are same in both the directions but in practical scenarios like road distances, distance between the two vertices would vary and the matrix becomes asymmetrical.

The TSP problem has been addressed by integer programming method in the literature. One of the earliest formulations was given by Dantzig et al. (1954). Dantzig and Ramser (1959) where they solve a 49 city problem. The formulation uses a binary variable x_{ij} equal to 1 if a optimal tour takes an edge (i,j) and the formulation is as follows:

minimize $\sum_{i<j} \mathrm{c}_{ij}\ x_{ij}$

Subject to:

1. $\sum_{i<j} x_{ij} + \sum_{j>k} x_{jk} = 2$

2. $\sum_{i<k} x_{ik} \leq |S| - 1 \qquad (S \subset V, 2 \leq |S| \leq n)$

3. $\mathrm{x}_{ij} = 0 or 1((i,j) \in E)$

Constraint 1 is flow balancing constraint which says whatever goes in has to come out and constraint 2 is called as sub tour elimination constraint where n is defined as number of vertices and V is set contains all the vertices. A tour is complete when it has all the vertices and comes back to the place where it started. Constraint 2 takes care of this part but the number of constraints become exponential as the problem size increases. This constraint makes the problem NP hard. Eastman (1958) developed a branch and bound algorithm for an asymmetric problem.Miller et al. (1960) gave a formulation in which he proposed the following effective sub tour elimination. C being the number of vertices.

$$\mathrm{u}_i - u_j + Cx_{ij} \leq C - 1 \qquad \forall (i,j) \in V \setminus 0, i \neq j)$$

$$\mathrm{d}_i \leq u_i \leq C \qquad \forall i \in V \setminus 0$$

This sub tour elimination is more effective and the number of constraints does not become exponential as the problem size increases. We will be using the sub tour elimination constraints of Miller et al. (1960) in our formulations in this report.

2.2 Vehicle routing problems (VRP)

The first paper with the phrase "Vehicle routing" in title is attributed to Golden et al. (1977).Vehicle routing problem (VRP) is a combinatorial optimization problem which designs optimal sets of routes to meet delivery of products to customers. Typical applications would be delivery of goods, waste collection, transportation, street cleaning, pick up and drop of students and employees etc. The VRP is a generalization of the abovementioned travelling salesman problem. In VRP we have set of vehicles, each with a fixed capacity to serve a set of customers, each having known demand. A vehicle will start at a node which is referred as depot, serves the subset of customers before returning to the depot and each fleet address only one route. TSP can also be seen as special case of vehicle routing problem where it has only single route and there is no capacity constraint.

Addition of the capacity constraint turns the assignment of vertices into a bin packing problem. Here the capacity of the vehicle is the bin capacity and the demands at the vertices are the items to be packed. The sum of the demands at the vertices visited by a vehicle cannot exceed the capacity of the vehicle.

The objective of a VRP would also differ based on the scenarios, but most commonly used objectives would be to minimize the:

1. Total distance travelled by all the vehicles

2. Total number of vehicles to be utilized

3. Total time travelled by the vehicle

4. Balancing the load of the vehicles

Recently researchers have used the objective of maximizing the quality of service. Though most of the work is based on a single or a composite objective, researchers have also considered multiple conflicting objectives. Jozefowiez et al. (2008) give an overview of the multiple objective routing algorithms in the VRP literature. His survey also indicates that the above four major objectives have been used in most of the multi objective VRP problems.

2.3 Classification of VRP

VRPs can be classified according to the following dimensions:

1. Types of fleet and Trips

2. Network

3. Nature of Demand

4. Types of operation

5. Operational constraints

6. Depot

Fig. 2.1 gives an overview about the classification and the highlighted ones are addressed in this research. The formulations and the case study consider VRPs with deterministic demand, pickup and delivery, homogenous vehicles and single trip. We consider commonly used objectives such as minimizing total distance and minimizing the number of buses.

The input data to the VRP is mostly deterministic and does not vary over the time. This becomes a huge challenge while addressing real life problems because demand and capacity can be dynamic. Researchers have also addressed this problem which is called as Stochastic vehicle routing problem (SVRP)

In this Literature survey chapter we address the following issues which are relevant to this research.

1. Solution methods to VRP

2. Heuristics and metaheuristics applications

3. Stochastic vehicle routing problems

4. Applications to delivering goods and employee pick up problems

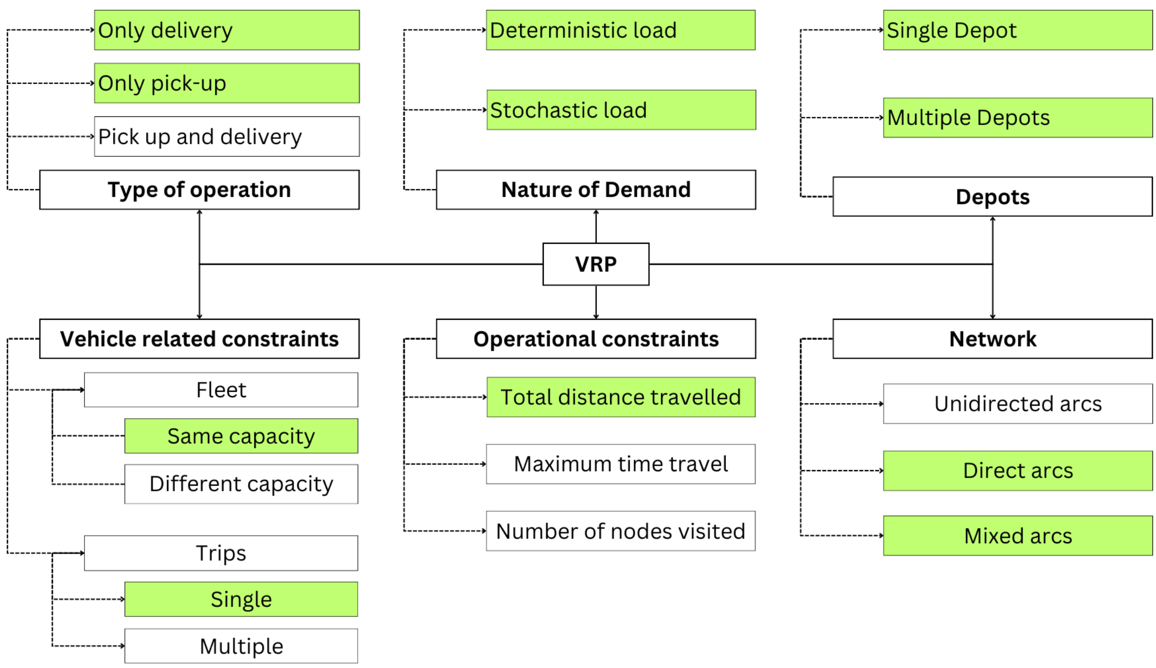


Figure 2.1: Classification of vehicle routing problems

2.4 Solution methods for VRP

Solution methods to VRP and its variants can broadly classified into exact methods and approximate methods. Laporte (1992) has classified algorithm into

1. Direct tree search method: it includes branch and bound, branch and cut, branch and price algorithms.

2. Dynamic programming: first seminal paper for VRP in dynamic programming was proposed by Eilon et al. (1971)

3. Set partitioning and column generation

4. Integer linear programming formulations

5. Heuristics

The report lists two direct tree search methods, a dynamic programming formulation Eilon et al. (1971) and three integer programming formulations. They also describe four popular heuristic algorithms including a Tabu search algorithm by Gendreau et al. (1994)

2.5 Heuristics

The exact methods are not capable of solving VRPs with large number of nodes in a reasonable time which is vital in real life problems. The literature addresses this problem through heuristics that sacrificing the optimality in order to get good solution within a reasonable solution of time.

The first heuristic algorithm for VRP was given by Clarke and Wright (1964) which has been applied in many practical situations. They computed a saving if two nodes were visited immediately by a vehicle and allocated nodes based on maximizing the savings. The first step of this algorithm is to assign a separate route for each customer. Iteratively the routes are selected and combined based on the potential for best saving in terms of cost. The following formula is used for the calculation of savings that can be gained by connecting two customers i and j: $s_{ij} = d_{io} + d_{oj} - d_{ij}$

Where 0 is the depot and dij is the Euclidean distance calculation mentioned previously. Many improvements have been suggested to the formula, Gaskell (1967) and Yellow (1970) gave a new formula: $s_{ij} = d_{io} + d_{oj} - \lambda d_{ij}$

Where λ - the shape factor which reduces the significance of distance to the depot and gives emphasis on the distance between i and j. Tyagi (1968) presents a method which groups vertices into tours based on the nearest neighbor concept. Vertices are added sequentially in such a way that the closest point is added to the last point. Gillet and Miller (1974) provided a sweep algorithm where rectangular coordinates for each vertex are converted to polar coordinates and used. Seeds are selected randomly with the depot as pivot and start sweeping from the seed to the depot. Polar angle is sorted in ascending order and we enlarge the angle before the point where the capacity constraint is exceeded. This provided better results when compared to the savings algorithm but at a higher computational cost. Other approach is to split the problem into two elements, clustering and routing which is called as cluster first, route second heuristic, given by Fisher and Jaikumar (1981).The first two methods become essential for this heuristics. Alternative to this heuristic is route first cluster second heuristics which was given by Beasley (1983). Beasley forms a single giant route as a first step. Then the routes are split into the appropriate number of individual routes. He uses Dijkstra (1959) algorithm for partitioning. The first heuristic for VRPB was given be Deif and Bodin (1984) and is an of extension of the Clarke and Wright heuristics. Sweep algorithm is extended by Solomon (1987) for the VRP variant with time windows.Nurcahyo et al. (2002) applied the sweep algorithm for generating public transport route.In the next thirty years after the work of Clarke and Wright, several exact and heuristics algorithms were proposed by researchers.The last thirty years has seen extensive use of meta heuristics and several new variants of the problem.Researchers have attempted to solve problems of large size as well as from practical applications. The static and deterministic version of this problem, the most basic version is called the capacitated vehicle routing problem (CVRP) which has been addressed extensively. Majority of the problems in real life scenarios have been addressed as CVRPs and its extensions, depending on additional constraints and problem assumptions.

2.6 Meta heuristics

The term meta heuristic was first introduced by Glover (1986) from the Greek words 'heuriskein' which means to find while the suffix meta means beyond. It is an iterative generation process that guides the heuristic algorithm by combining intelligently different concepts for exploring and exploiting the search space.Learning strategies are used to structure the information in order to find efficiently near optimal solutions Osman and Laporte (1996). The meta heuristic algorithm needs a seed to start with, any one of the above-mentioned heuristics will be employed to generate initial seed. Some methods create multiple solutions which will be needed for some meta heuristics algorithm like genetic algorithm. Metaheuristic algorithms are highly efficient for most of the combinatorial optimization problems, particularly it is effective for VRP problems and its variants. The objective is to explore a larger solution space even by allowing the infeasible solutions in order to find the near optimal solution. These Meta heuristics are not problem specific and can be classified based on population, local search and learning mechanism. Over the last thirty five years the following meta heuristics and many more have evolved BoussaïD et al. (2013)

1. Simulated Annealing
2. Genetic Algorithms
3. Tabu Search
4. Ant Colony Optimization
5. Particle Swarm Optimization
6. Bee Colony Optimization
7. Biogeography-based Optimization

The broad classification is based Single solution based methods and Population based meta heuristics. While Simulated Annealing and tabu search come under single solution based methods, the rest are under population based. The above methods have several variants that have been tested on a number of combinatorial optimization problems and different variants and implementations are found to be effective for different problems.

We describe Genetic Algorithm and tabu search in detail, since we use these two techniques in this thesis.

The following methods are used widely and we use these to solve our research problem.

2.6.1 Tabu Search

The Tabu search concept was proposed by Glover (1986).The basic principle of this algorithm is to repeatedly make neighborhood to explore the search space and improve the solution. It has a novel future which prevents the algorithm to cycle back to the previously visited solution by using the Tabu list that records the recent history of the search. It enhances its performance by accepting the no improving solution so that the search doesn't stuck at the local optimum. The first application of Tabu search on the vehicle routing was given by Osman (1993)

Gendreau et al. (1994)) developed a tabu search based algorithm to solve the VRP with capacity and route restrictions. The algorithm considers a adjacent solutions obtained by repeatedly removing a vertex from its current route and reinserting it into another route. They use a generalized insertion procedure previously developed by them. Infeasible solutions are permitted with a penalty. Numerical tests on a set of benchmark problems indicate that their tabu search algorithm outperforms the best existing heuristics.

Cordeau and Maischberger (2012) propose a parallel iterated tabu search heuristic for solving four different routing problems: the classical vehicle routing problem (VRP), the periodic VRP, the multi-depot VRP, and the site-dependent VRP. Their algorithm is also applicable to situations involving time windows. Their computational results show that the proposed algorithm competitive with recent heuristics designed for each of the problem studied.

Xia and Fu (2019) have developed an improved Tabu search for open VRP considering hard and soft time windows. The two objectives considered are minimization of number of vehicles and total cost, which includes a penalty for deviation from time windows. The tabu search algorithm considers five transformations between two randomly chosen customers. They have tested the algorithm on problems with up to 100 nodes

2.6.2 Genetic Algorithm

Genetic algorithm is a population based meta heuristics algorithm that follows the idea of biological evaluation and natural process where the fittest individual will survive. Key aspects of this algorithm involve combining and mutation of the solution. The whole process of the genetic algorithm is described following steps.

1. Initial solutions are created by any of the constructive heuristics. Representation of the solution is different for each problem.

2. Fitness functions are defined and estimated for the initial solutions.

3. Subset of the solution from the populations are selected for the crossover.

4. Define the type of crossover and apply it to the selected subset.

5. Define and apply the type of mutation operator at its associated probability.

6. The fitness function values are evaluated and the solution with the worst value is removed.

7. Stop if the stopping criterion is met or go to step 3.

The chromosome is the representation of a problem solution.

Several researchers have worked on solving the capacitated VRP and its variants using genetic Algorithms.Baker and Ayechew (2003) developed a genetic algorithm for the basic VRP and showed that the results are comparable to that using Simulated Annealing and tabu search. Berger and Barkaoui (2003) develop a hybrid genetic algorithm for the basic VRP. They mention that GAs have been shown to perform well for VRP with time windows. They use two populations to minimize total distance travelled. Their results indicate that the proposed algorithm is very competitive compared to best known algorithms. More recently, da Costa et al. (2018)) propose a genetic algorithm to minimize CO_2 emissions in the route. They use road speed and gradient data. They are able to show reduction in emission without increase in cost. They have used google maps to compute distances while solving a real life case study

2.7 Stochastic vehicle routing problem (SVRP)

2.7.1 Demand stochasticity

The classical Vehicle Routing Problem (VRP) involves optimizing routes for multiple vehicles to visit a set of customers while adhering to various constraints. Stochastic Vehicle

Routing Problems (SVRPs) introduce randomness into elements like demand and travel times, challenging traditional solution methodologies. SVRPs are often modeled using stochastic programming, with solutions determined in two stages: a planned solution in the first stage and corrective actions in response to realized random variables in the second stage. Chance constrained programs (CCPs) and stochastic programs with recourse (SPRs) are common modeling approaches, with recourse policies designed to address route failures caused by uncertain demand. While exact algorithms exist for some SVRPs, most approaches rely on adapted heuristics.

The Stochastic Vehicle Routing Problem (VRP) is a widely studied optimization challenge in logistics and transportation management, especially in scenarios where customer demands vary unpredictably. Traditional methods for addressing stochastic demands often rely on heuristics or metaheuristic algorithms. One approach, as outlined in A vehicle routing problem with stochastic demand by Dimitris J. Bertsimas, involves initializing routes based on historical demand patterns. Vehicles are dispatched along these routes, with demand at each node becoming known as the vehicle progresses. If capacity constraints are exceeded, the vehicle returns to the depot to start a new route. This method leverages historical data to guide initial routing decisions while adapting to dynamic demand variations in real-time.

Another variant of the Stochastic VRP considers uncertainties not only in demand but also in travel and service times given by Guoming Li AND Junhua L, along with soft time windows for deliveries. In this approach, the objective function aims to minimize the total cost, including travel costs and penalty costs for expected early or delayed arrivals. Arrival times are modeled as normal distributions, allowing for the calculation of expected values for deviations from the time windows. Tabu search, a metaheuristic algorithm, is applied to explore sequences of nodes, with preference given to nodes closer and with shorter time windows.

Recent research by Zangir Iklassov, Ikboljon Sobirov in Reinforcement learning approach to stochastic vehicle routing problem with correlated demands has explored the application of Reinforcement Learning (RL) techniques to address the Stochastic VRP with correlated demands. This approach involves encoding various inputs such as weather information, customer locations, and dynamic demand into a model that generates state embeddings for each customer. The vehicle's current position and capacity are encoded as memory embeddings. These embeddings are then combined through an Attention Layer to obtain probabilities over nodes, representing the likelihood of each node being the next position of the vehicle. RL algorithms are trained to learn optimal routing policies based on the observed environment states, enabling adaptive and dynamic decision-making in response to stochastic demand patterns.

2.7.2 Extension to employee pick up problems

The uncertainity is internal and external factors have led to development of stochastic approaches to manage employees which involves problems related to pickup, scheduling, etc. This section highlights the major applications for the same.

Ming Liu, Bian Liang wrote about assigning airline check-in employees to tasks related to departing flights at an international terminal of a large airport, subject to uncertainties stemming from flight delays and traffic control in the paper titled Stochastic Check-in Employee Scheduling Problem. The objective is to minimize both staffing costs and the risk associated with time uncertainty. To address this, two models are proposed: a two-stage stochastic model based on expectations and a risk-averse model incorporating conditional value at risk (CVaR). Heuristic approaches are employed to solve these complex models efficiently. The sample average approximation method is utilized to approximate the expected objective function by calculating the sample average from Monte Carlo samples, enabling the formulation of a deterministic program solvable with commercial solvers. Additionally, the problem is decomposed into scenario sub-problems, allowing for iterative updates of solutions and dual prices to converge to near-optimal solutions, particularly effective for handling large-scale instances through parallel processing.

Another paper titled People Scheduling Service: an Employee Scheduling Algorithm based on Stochastic Workloads, a master thesis discusses about Picnic Technologies facing the challenge of scheduling employees efficiently for their People Scheduling Service, catering to the unpredictable demands of a supermarket environment. With grocery demand and workload fluctuating due to customer preferences, seasonal changes, and unforeseen events, the objective is to minimize expected costs while considering factors like lost revenue, excess personnel costs, and employee satisfaction. Two models are proposed: mixed integer linear programming (MILP) and an L-shaped algorithm. The MILP directly solves the scheduling problem but may encounter scalability issues. On the other hand, the L-shaped algorithm, a two-stage stochastic optimization approach, addresses employee satisfaction and scheduling constraints through master and subproblems. By generating robust schedules through subproblems for various scenarios, this algorithm offers a practical solution.

This furthers extends to The Stochastic Electric-Vehicle Relocation Problem with Dynamic Pricing which addresses the challenges of optimizing vehicle distribution and profits in flexible car-sharing services. The SE-VReP-DP model, a two-stage linear stochastic mixed-integer programming approach, integrates decision-making on vehicle relocation and pricing strategies. In the first stage, decisions on vehicle relocation and pricing are made without knowing customer preferences. The second stage considers customer behavior scenarios and preferences. Stochastic customer behavior is approximated using Sample Average Approximation (SAA), with a Random Utility model handling uncertainty in preferences.

Another study by Youngbum Hur, Jonathan F. Bard in A stochastic optimization approach to shift scheduling with breaks adjustments talks about addressing the challenge of scheduling shifts and breaks for service industry workers under uncertain demand, this study employs a two-stage stochastic optimization approach. In the first stage, shifts are assigned without precise knowledge of demand, shifts are assigned to workers without knowing the exact demand for each time period, representing proactive decision-making under uncertainty. While in the second stage, decisions are made based on observed demand, representing a reactive approach to decision-making. Each scenario represents a daily demand realization with equal probabilities. The Recourse Problem (RP) minimizes expected costs under the true demand distribution, providing the optimal "here-and-now" solution for stage 1. The Wait-and-See Solution (WS) solves sub-problems by scenario, assuming perfect future demand information, offering insights into optimal decision-making under uncertainty for stage 2.

In conclusion, the need for employees in a factory is influenced by:

1. Factory needs: Demand for that particular month or shift of the day

2. Employee needs: Certain employees might be on leave owing to personal reasons or external reasons, the factory operations should be running even then and re-

scheduling with available employees need to be done

These factors influence the number of employees to be picked up from each node. The problems and algorithms used in these papers can be extended to the employee pick up problem referring to our particular case.

2.8 Conclusion

The purpose of this chapter is study the previous work done on the research questions that we seek to address in this thesis and find motivation from the literature for the problem under study. This literature review gives a detail view on the evaluation of the vehicle routing problem starting from the travelling salesman problem and discusses the classification of the vehicle routing problem to finally discussing the stochastic VRP. The Second part of this chapter discuss the three special variants of the VRP in detail which is in line with our problem definition. Though certain aspects of the problem under study have been addressed in the literature, two important aspects to the problem have not been addressed and make our problem unique. These are:

1. The consideration of multi-depot system for vehicles, this is very common given most of the factories outsource the employee pick up service.

2. Introducing stochasticity for the employee pick up demand at each node after factoring in the various possibilities for uncertainity in employee demand

In the following chapters, employee pick-up is addressed using different models and implementing more realistic constraints as we progress further in the report. This increases the problem complexity but also provides a better solution to the problem.

Chapter 3

VRP: Formulation and heuristics for employee pick up problem

3.1 Problem Under Study

We address the employee pick up problem under different parameter settings. Parameters/ terminologies and definitions:

- Node: A stop where a group of employees are waiting to be picked up
- Number of nodes: Models are tested against different number of nodes to understand the robustness, helps in further model optimisation
- Depot: Place where the vehicles start and end their journey from
- Number of depots: We start by using 1 depot i.e. vehicles start and come back to the factory, in further formulations multiple depots have also been considered
- Demand at each node: Number of employees waiting at each node, this varies to understand the robustness of the algorithm
- Vehicle capacity: The maximum capacity a vehicle can hold
- Number of vehicles: Given as per the problem statement, is varied to understand the model robustness

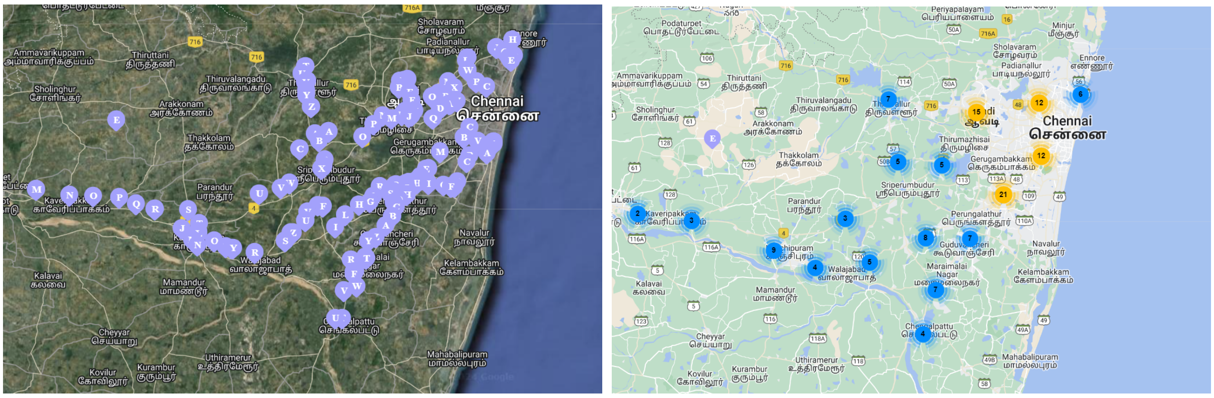
The demand (or the number of employees getting into the vehicle or exiting from the vehicle) is different for each node(point). In single depot VRP problems it is customary to assume that all the vehicles start from the depot and end at the depot. In employee pick up the final destination is the depot (factory or office) where employees arrive. Thus following problem statements are studied:

1. Vehicles start from single depot which is also the factory picks up employees across all nodes and returns back to the factory
2. Vehicles start from multiple depots, picks up employees across all nodes and return back to a common endpoint which is the factory
3. Stochasticity:

- Problem statement shift to demand stochasticity to understand applications of delivering goods to nodes with varying demands using a vehicle with known capacity
- Same understanding is extended to employee pick up problem to understand its implications within our context

The problem setting and assumptions are indicated as follows:

- A homogeneous fleet of vehicles with fixed capacity are used.
- Travel distance is used as costs for minimising
- External factors like weather/ road conditions are completely ignored
- Distances are calculated based on geographic coordinates
- Distance between nodes is symmetric i.e. Distance(N1, N2) = Distance(N2, N1)



(a) Hybrid view of individual nodes

(b) Roadmap view of clustered nodes

Figure 3.1: Geographical node locations for real data

3.2 Model 1: Pulp Model

3.2.1 Introduction

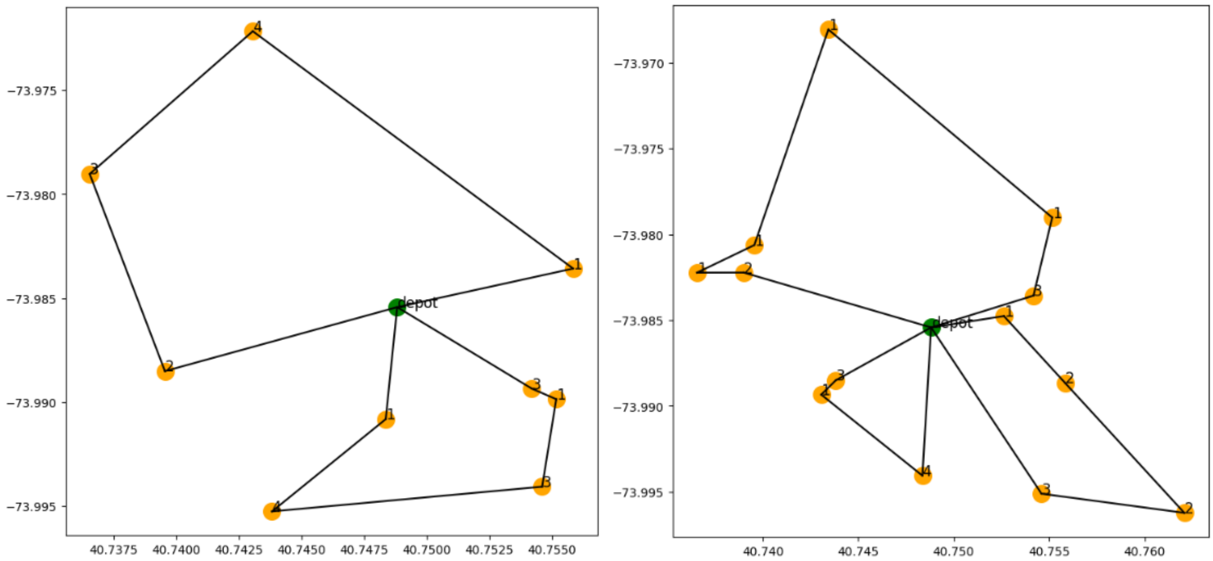
It is an inbuilt model which uses COIN OR Branch and cut solver (CBC) open source mixed integer programming (branch and bound) solver written in C++. Distance for this model is euclidean. It has low computational power so works well only for small datasets.

3.2.2 Algorithm

Decision variable: $X_{ij}^{k} = \begin{cases} 1, & \text{if the vehicle } k \text{ goes from location } i \text{ directly to location } j \\ 0, & \text{otherwise} \end{cases}$

Minimize: $\min \sum_{k \in K} \sum_{(i,j) \in E} c_{ij} \; x_{ijk}$

The minimisation function is trying to minimize the sum of travelling cost for all vehicles



(a) 10 nodes, 2 vehicles

(b) 14 nodes, 3 vehicles

Figure 3.2: Pulp model plots

Subject to:

1. $\sum_{k\in K}\sum_{\substack{i\in V\\ i\neq j}} \mathrm{x}_{ijk} = 1 \qquad \forall j \in V \setminus \{0\}$

2. $\sum_{i\in V\setminus\{0\}} \mathrm{x}_{0jk} = 1 \qquad \forall k \in K$

3. $\sum_{\substack{i\in V\\ i\neq j}} \mathrm{x}_{ijk}$ - $\sum_{i\in V} \mathrm{x}_{jik} = 0 \qquad \forall j \in V \quad k \in K$

4. $\sum_{i\in V}\sum_{\substack{j\in V\setminus\{0\}\\ i\neq j}} \mathrm{q}_i \; x_{ijk} \leq Q \qquad \forall k \in K$

5. $\sum_{k\in K}\sum_{\substack{(i,j)\in S\\ i\neq j}} \mathrm{x}_{ijk} \leq |S| - 1 \qquad S \subseteq V \setminus \{0\}$

6. $x_{ijk} \in \{0,1\} \qquad \forall k \in K \quad (i,j) \in E$

The above constraints help define the problem as follows:

1. Only 1 visit per vehicle per customer/ node location is permitted
2. All vehicles have to depart from the depot which is node 0
3. Number of vehicles coming in and going out of any particular node is the same
4. Delivery capacity of any vehicle should not exceed it's maximum capacity
5. Removal of subtours (A subtour is defined as a route which does not start and end at the depot)

6. Decision variables

3.2.3 Limitations

One major limitation of this model is the computation cost, for 10 nodes and 2 vehicles the model gives instantaneous results. But if the node points are increased to 15 nodes it takes 9 minutes to finish running completely.

Thus it is evident that unless you have higher computational machines using this model for larger problems is highly inefficient.

3.3 Model 2.1: Tabu Search

3.3.1 Introduction

Tabu Search is a metaheuristic algorithm used for solving combinatorial optimization problems. It iteratively explores neighbouring solutions with the objective of finding an optimal or near-optimal solution. The algorithm maintains a memory structure (the "tabu list") to avoid revisiting recently explored solutions, thus promoting diversification and preventing premature convergence to local optima.

3.3.2 Algorithm

It operates within a predefined set of nodes, typically determined by the initial solution. It explores neighbouring solutions through iterative exchanges of customers between routes, guided by a set of rules and constraints. The algorithm maintains a tabu list to prohibit revisiting recently explored solutions, thereby promoting diversification and preventing cycling.

Initial solution determination using NNH (Nearest neighbour heuristic)

The most popular heuristic for the vehicle routing problem is the nearest neighbour heuristic. In this algorithm the rule is to go to the nearest unvisited point inclusion of which does not exceed the vehicle capacity. If the capacity of the vehicle is exceeded, we create a new vehicle and continue till all the nodes are assigned to the vehicles. There is a randomness associated with the solution because the next depot from which a vehicle starts is generated randomly. Nearest neighbour heuristic is simple but it sometimes misses the shortest route that a vehicle can take, and hence is a greedy heuristic. Every solution that is generated by the heuristic should include all the points and vehicle capacity should not be exceeded.

The steps to generate initial feasible solutions are as follows:

Step 1 Select a depot randomly from the depot set and start a new route.

Step 2 Find the shortest distance from the selected depot to each boarding point.

Step 3 If the number of employees to be picked in the nearest boarding point is less than the capacity of vehicle then;

Add the boarding point to the new route

Else;

Add find next nearest boarding and go to step 3

Step 4 Repeat the process until the capacity of the vehicle is exceeded or no more points can be added.

Step 5 Go to step 1;

Step 6 Repeat until the depot set becomes zero.

We do not start a new route until the current vehicle is full which makes all the boarding points to be accommodated within the available buses.

Post calculating the NNH, Tabu search model is applied which performs node exchanges within a route provided by the NNH. These iterations give us the final minimal solution.

Assumptions:

- Distance Calculation: The distance between locations is estimated using the Euclidean distance formula adjusted by the latitude to approximate real-world distances. The use of an API for precise distance calculation is considered but requires further investigation.
- Capacity Handling: The algorithm allows for flexibility in handling vehicle capacity constraints. If a vehicle's capacity is exceeded, a new route is initiated, ensuring compliance with capacity limitations.
- Initial Solution Generation: The initial solution is constructed iteratively by assigning each vehicle to the nearest unvisited customer until the vehicle's capacity is exceeded. This method, akin to the Nearest Neighbor Heuristic (NNH), influences the efficiency and effectiveness of the Tabu Search

Key Differences from NNH:

- Capacity Handling: Unlike NNH, which may overlook capacity constraints, Tabu Search explicitly accounts for vehicle capacity limitations by breaking routes when necessary.
- Route Assignment Logic: Tabu Search employs a modified greedy approach for route assignment, integrating capacity checks to optimise solution quality.
- Number of Vehicles: While NNH may allow for a variable number of vehicles, Tabu Search assigns customers to a fixed number of vehicles, each with its own route. This ensures better control over fleet management and resource allocation.

3.3.3 Limitations

- Algorithm related: For routes where each vehicle serves only two nodes, the initial solution may coincide with the final solution due to the symmetry of distance calculations. The limitation is addressed in the next model. E.g. For initial solutions where a route comprises only two nodes, such as [N1, N2], this configuration remains unchanged in the final solution.
- Parameters: The performance is influenced by parameters such as the number of iterations (Num_iterations). Adjusting this parameter can affect the balance between solution quality and computational efficiency. Higher numbers of iterations allow for more extensive exploration of the solution space but may increase computational overhead.

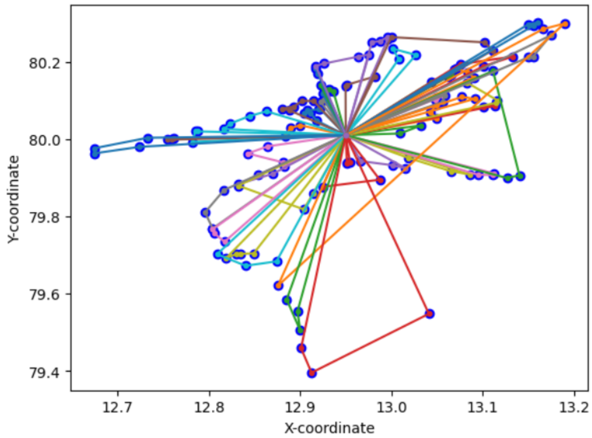
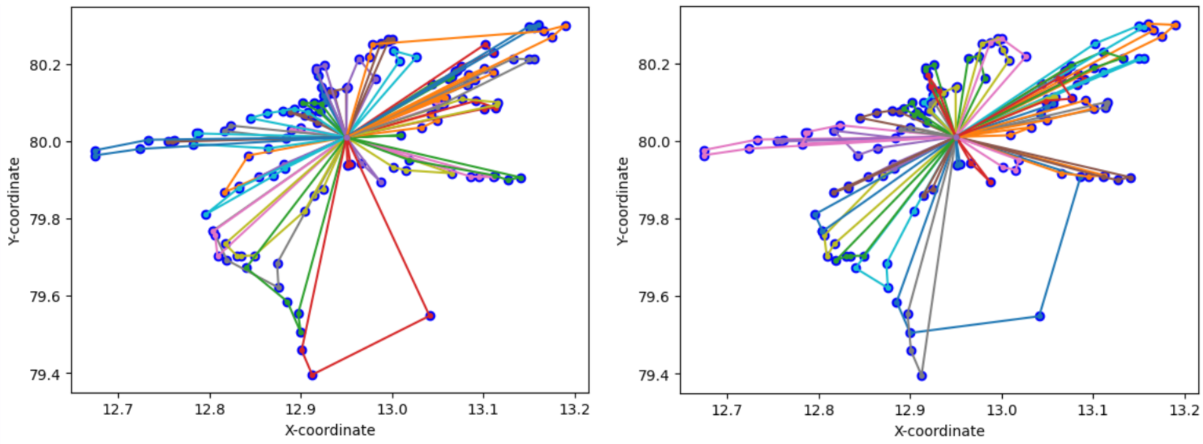


Figure 3.3: Best distance: 2469

3.4 Model 2.2: Tabu Search - Modification to 2.1

3.4.1 Algorithm



(a) Best distance: 2210, initial solution is random

(b) Best distance: 2193, initial solution is genetic model solution

Figure 3.4: Tabu search model comparisons

We use the nearest neighbour heuristic solution (NNH) as the initial solution as the quality of the Tabu search solution depends on the initial solution to a certain extent. Different types of neighbourhood moves are applied to the current solution. These are explained below: Two different vertices are selected randomly from the current solution and one of the four mechanisms based on 2opt is applied. The 2 opt algorithm is well known. It was introduced by Lin and Kernighan (1973) and has been used in Osman (1993). Brandão (2004) uses two types of trailing moves which are swap move and insert move. We incorporate these moves in our Tabu search algorithm. The moves used in our algorithm are:

1. *Vertex Reassignment*: The first selected vertex is removed from its current position and inserted into the position before the second selected vertex.

2. *Vertex Swap*: Positions of two selected vertices are interchanged.

3. *2-opt*: All elements between two selected vertices are interchanged

4. *Tail Swap*: The tails after two selected vertices are swapped till the end of the selected route.

These moves will make the new solution infeasible but allows the Tabu search to explore more solutions and improves the chance for the mechanism to give us a better Solution.

For the first objective we need to minimise the total distance travelled by all the vehicles which is the travelling cost of all the employees. We add a penalty value to the objective function when vehicle capacity is exceeded. The evaluating criterion for the Tabu search algorithm is given by:

$\sum_{k=1}^{K}[d(k) + p(E(k))]$

Here K is the total number of vehicles, $d(k)$ is the distance travelled by the k th vehicle, $E_l(K)$ is the excess load in the vehicle k and p is the penalty coefficient for not meeting the capacity constraint which is defined by the user.

The Tabu search algorithm is run for a prespecified number of iterations counted from the last best solution. Interval of p could be between 0.0001 and 200000 and is normally equal to 1 by default. This mechanism was used in the algorithm of Gendreau et al. (1994). The Tabu list consists of moves which are attributed to the last five to ten moves in the iterations. In case of vertex reassignment if the vertex (i) is selected in the particular move then the element i is saved in the Tabu list. Element (i, j) in case of other three neighbourhood moves will be saved in the tabu list. The tabu list is updated iteratively by updating the latest moves and removing one from the list. This is to ensure the algorithm does not get trapped into any of the previous best solution found. The search will be terminated based on two criteria a total pre-specified number of iterations or number of iterations since the last best solution.

Iter: current number of iterations

Max_iter: maximum number of iterations

Cons_iter: current number of consecutive iterations without any improvements in the best solution that we got

Max_cons_iter: maximum number of consecutive iterations without any improvements in the best solution

Cand_list: current number of vertex candidate moves on the list

max_cand_list: maximum number of candidates moves on the list

The tabu search algorithm is as follows: Generate a feasible solution by using the nearest neighbour heuristics or take best feasible solution that genetic algorithm has been generated

Initialise iter and cons_iter with zero values

While ($iter \leq max\_iter$) and ($cons\_iter \leq max\_cons\_iter$) do

while ($cand\_list \leq max\_cand\_list$) do

Select two vertices randomly

Select one of the four types of neighbourhood move randomly

Add the solution produced by the selected move to candidate list

End;

Select the best solution in the candidate list if it is not Tabu or it produces a solution strictly better than the best solution so far

Set the new solution as the current solution, update the tabu list and increment iter

If the new solution improves the best solution so far
Update the best solution so far and set *cons_iter* to 0 End.
This algorithm is run for multiple runs to converge at the best possible solution.

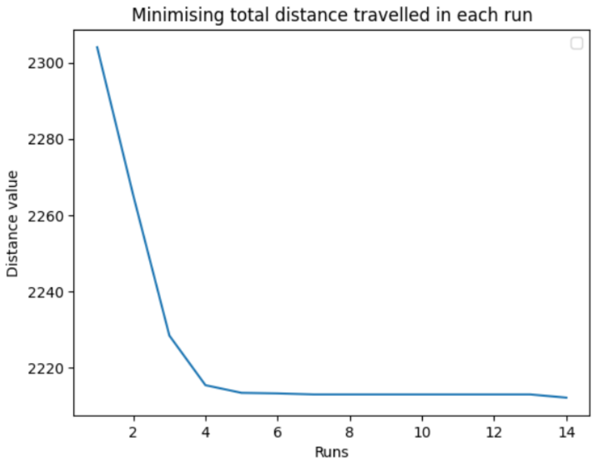


Figure 3.5: Converging to the solution

3.5 Model 3: Genetic

3.5.1 Introduction

The proposed genetic algorithm aims to give a high-quality solution within a reasonable time. The initial population of the chromosomes can either be generated randomly or using heuristic solutions. The performance of any Genetic algorithm depends on the initial solution and the evolutionary techniques maintains a balance between the exploration and exploitation. The genetic algorithm is developed to address problems of larger size.

3.5.2 Algorithm

Chromosome representation

The chromosomes are represented reflecting the properties of the problem under Consideration. For e.g. Let us assume a solution [[1,2],[3,4,5],[6,7]] - this means there are 7 nodes and 3 vehicles where vehicle 1 picks up employees from nodes 1 and 2. For the above problem, in the context of genetic algorithm, the solution itself is the parent. Individual boarding points like 1,2,3, etc. are called the genes. The genes together making a sublist e.g. [1,2] is called a chromosome. So the parent has 3 chromosomes and 7 genes.

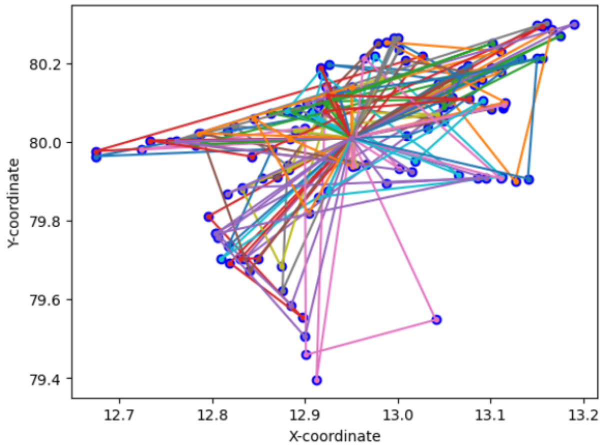
Initial population (Mating pool)

As initial population is an important factor that determines the solution, 2 methods are devised based on this to implement the model:

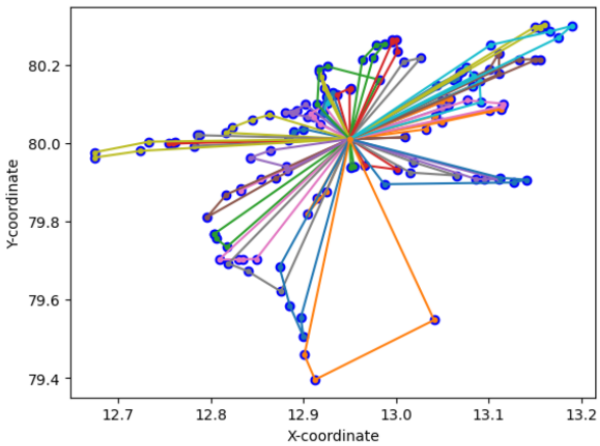
1. Tabu search solutions: The solutions generated in Tabu search algorithm i.e. Model 2.2 which are $>$ 3 times the optimal solution are selected in the mating pool

2. Randomly generated population: Using the NNH algorithm random initial solutions are generated. These reproduction techniques will make the solution infeasible and may exclude some of the points. We repair each route by checking the applicable capacity and distance constraints. We reformulate the route with nearest neighbour heuristics within each route.

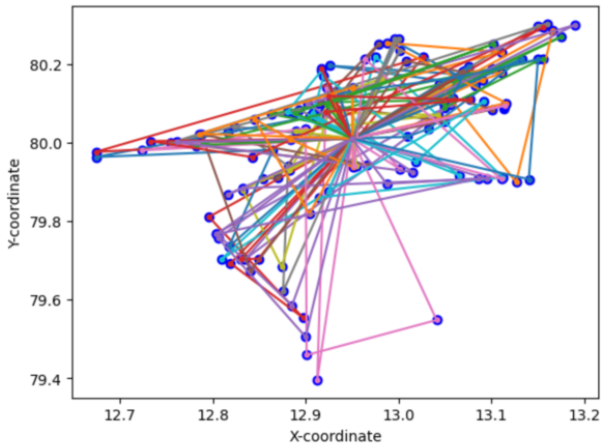
The 3rd method is developed by using the inbuilt Python PyGad Library



(a) Best distance: 4409, initial population is randomly generated



(b) Best distance: 2597, initial population from Tabu model using PyGad library



(c) Best distance: 2210, initial population from Tabu model using Genetic Model

Figure 3.6: Genetic model comparisons

Reproduction process We use two point crossover and mutation as reproduction processes. The two new solutions produced after the two point cross over can be infeasible either because of capacity restrictions or it would not include all the boarding points in the solution (some of them could repeat). In such scenarios we reconstruct the solution by using the nearest neighbour heuristic again for each vehicle to form new route with the left out depots and boarding points. The mating pool considers only feasible solutions which includes all the boarding points. Two point crossover and mutation is used to generate offsprings. The basic principle for two point crossover is as follows:

Step 1: select 2 parents at random from the initial population

Step 2: select 2 points randomly for exchanging the chromosomes within that range

Step 3: Perform the exchange apply the reconstruct operation to convert the solution into feasible

The performance of the genetic algorithm is further enhanced by including the mutation process. The basic principle is two genes are selected randomly; they can be from one chromosome or 2 different chromosomes and their positions are exchanged. The steps are:

Step 1 select genes at random for the mutation

Step 2 change the vehicle number randomly, which is done by two ways explained below which are selected randomly for each iteration:

- Two genes are selected at random in the offspring solution. The first selected gene is removed from the solution and reinserted into the position before the second selected gene.
- The second way is to swap places of the two selected genes. The mutation process can also make the solution infeasible when the capacity is exceeded. We add a large penalty to the fitness function so that the infeasible solution will not be considered in the process itself.

The steps of the Genetic algorithm are described below Generate n initial population of solutions

Evaluate the fitness value of each solution

Repeat

Select the two parents at random from the population

Produce two offspring using the 2 point cross over function or mutation function

Choose the best offspring evaluated by fitness function with penalty

Check whether the offspring fitness has improved if yes update the best solution

Check whether the offspring fitness is less than either of the parent fitness if yes

Select the max fitness parent in the population to be replaced by the offspring

End if the stopping criteria is attained

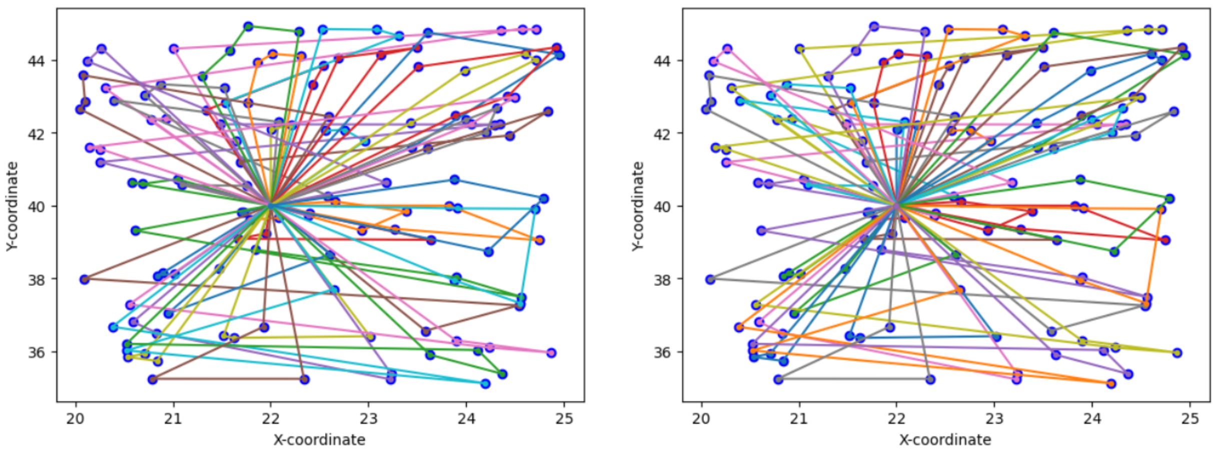
If both the mutated offspring aren't less than the parent fitness values then keep the parents in the initial population and proceed to next iteration

The stopping criteria is defined as the specified number of iterations and we vary based on the problem size

Fitness Function The objective is to minimise the total distance travelled by all the vehicles. It also penalises the vehicles that carry more than their capacity. The fitness function is $\sum_{k=1}^{K}[d(k) + p(E(k))]$

3.6 Model Performances

Comparing Genetic Algorithms (GA) and Tabu Search, GA often took longer to converge than Tabu Search. However, Tabu Search consistently delivered superior results, especially when initialised with GA solutions rather than the Nearest Neighbor Heuristic (NNH). Conversely, GA benefited from integrating optimal solutions from Tabu Search into its genetic population. These findings emphasise the synergistic potential of combining different optimization techniques for robust performance across diverse problem



(a) When GA and Tabu search gave the same optimal solutions, dataset 5

Dataset No.	Tabu Search Soln.	Genetic Algo. Soln.
Original Dataset	2210	4421
data_1	20372	45066
data_2	40980	119453
data_3	56207	96482
data_4	32080	78741
data_5	50809	50809

(b) Model performances across datasets

Figure 3.7: Model comparisons

domains. This was compared to 5 different datasets with similar data size, model priorities stayed mostly the same.

Overall Tabu search model with Genetic algorithm solution being used as the initial solution gave the best result.

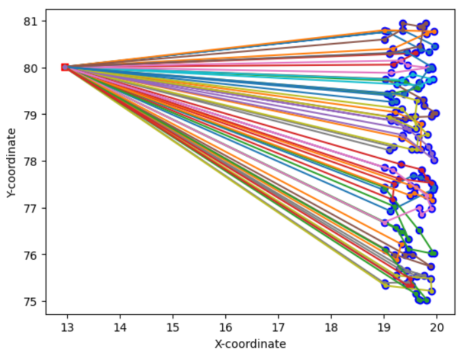


Figure 3.8: Interesting plots when depot is far away from employee nodes

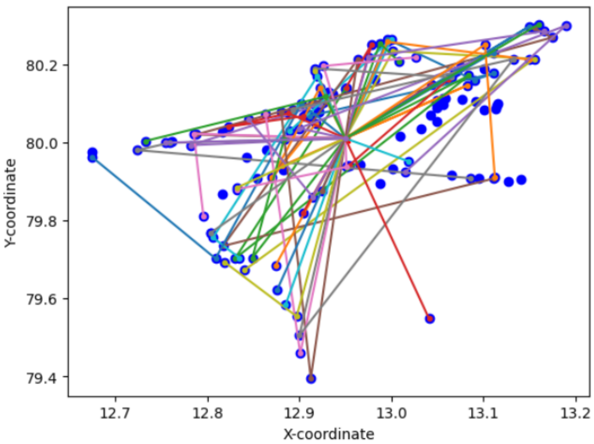
3.7 Extending to Multi-Depot

3.7.1 Introduction

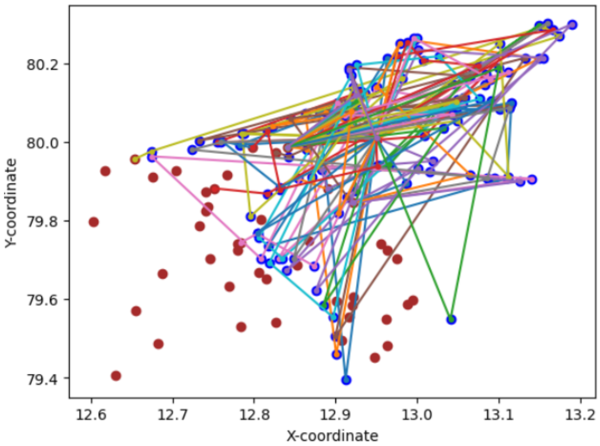
The only difference for this case is that the number of starting depots are equal to the number of vehicles. While the end point for each vehicle is a common depot or factory. There are 2 variations for the above:

- The starting depots are the first nodes on the route.
- The starting depots are randomly generated points in the existing boarding points geographic landscape. Each vehicle is free to choose the best possible depot such that the total route cost is minimised for its journey.

It is very evident that the 1st variation will give a lesser cost as compared to the 2nd because the initial cost of travel i.e. from starting depot to the first node is saved. The following plots provide a glimpse of the same.



(a) Best distance: 4131, starting depots are the first nodes on the route



(b) Best distance: 4639, starting depots are randomly generated points

Figure 3.9: Multi depot comparisons

```
Genetic Algorithm instance used for the model
ga_instance_multi = pygad.ga(num_generations=10,
                             num_parents_mating=10,
                             fitness_func=fitness_func_multi,
                             sol_per_pop=5,
                             num_genes=num_nodes,
                             gene_type=int,
                             parent_selection_type="tournament",
                             k_tournament=3,
                             crossover_type="two_points",
                             crossover_probability = 0.8,
                             initial_population = custom_initial_population,
                             mutation_type="swap",
```

```
                    mutation_percent_genes=10,
                    mutation_num_genes=2,
                    mutation_probability = 0.8,
                    max_val=2)
```

Chapter 4

Stochastic VRP

4.1 Introduction

We make a slight change in the problem statement to implement stochasticity better before extending it to our original employee pick up problem. In the problem scenario, each node represents a customer with an associated demand, which follows a probability distribution such as Normal or Uniform. However, the demand at each location remains uncertain at the time of route planning. Consequently, designing routes based on known demand becomes impractical due to various constraints:

- Limited resources hinder the ability to accommodate dynamic demand variations efficiently
- The effort required to continuously adjust routes may outweigh the benefits gained from precise demand forecasting.
- Ensuring regularity and personalization of service, such as maintaining the same vehicle and driver for specific customers, takes precedence over demand certainty.
- The challenge of learning demand patterns a day before customer visits adds complexity to route optimization strategies

This problem finds applications across diverse sectors, including central banking for the collection of funds from branches, post office operations for package distribution, retail logistics for managing demand at stores and warehouses, and healthcare logistics for optimising service delivery to various facilities. In each context, the uncertainty surrounding customer demand necessitates flexible and adaptive routing strategies to ensure efficient resource utilisation and satisfactory service levels.

This understanding can be extended to the employee pickup problem statement as well because there are a lot of stochastic parameters which have been considered as constant or ignored for formulating the previous algorithms. In a real world scenario traffic accidents, weather changes, road congestion, and other factors will affect the original distribution plan, so it is more practical to study the stochastic vehicle routing problem (SVRP)

Variables:

- Number of vehicles
- Vehicle capacity

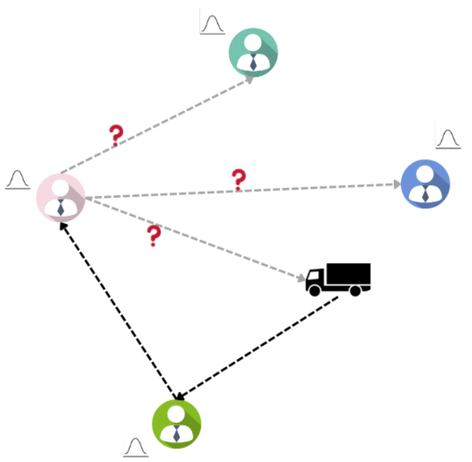
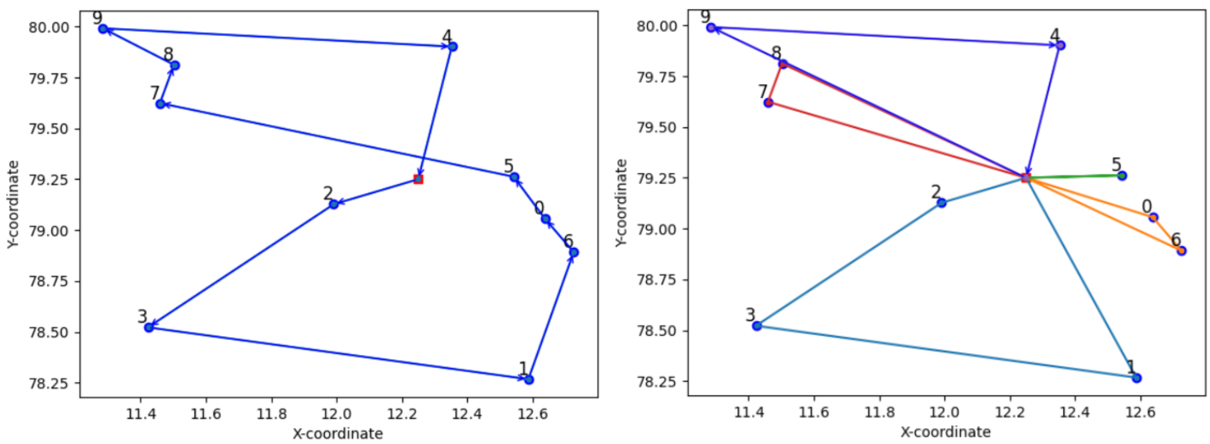


Figure 4.1: Stochastic Vehicle Routing

- Node points
- Depot
- Demand distribution

4.2 Model 1: Demand based routing



(a) Priori customer sequence

(b) After introducing stochasticity

Figure 4.2: SVRP: Demand based routing

In the context of managing uncertain customer demand in vehicle routing, establishing a fixed a priori sequence among all customers is essential for efficient route planning. This sequence is derived from previously known demand data, typically utilising statistical metrics such as the most occurring demand value or the mean demand value. By identifying the most frequent demand value, the sequence prioritises customers with higher demand frequencies, aiming to streamline the routing process. Similarly, computing the mean demand value provides insight into the average demand level, guiding the arrangement of customers along the route for optimised resource allocation and service delivery.

The vehicle has to travel to the depot to refill if it cannot meet the demand of the next node.

4.3 Model 2.1: Expected Value with scenario probabilities

4.3.1 Introduction

Now we approach the stochastic problem with an expected value point of view. We use the expected value property of probability and statistics to help us determine the best sequence in which a vehicle should travel to get minimal costs under any scenario. The different scenarios and probabilities of achieving each are provided, this information could be based on past data analysis. As the number of scenarios and their corresponding probabilities increase we get a better solution which produces optimal results under any stochastic circumstance. This approach has been used in paper *Solution Algorithm for the Vehicle Routing Problem with Stochastic Demands* by *Ryota Omori* and *Takayuki Shiina*.

Variables:
Total number of scenarios: n
i^{th} scenario: S_i
Probability of i^{th} scenario: P_i
Total number of node permutations: m
Travel cost for a permutation: C_j
Additional cost for a permutation given a demand scenario: c_{ij}

Objective: Minimise: $C_j \sum_{i=1}^{n} P_i \ c_i$

4.3.2 Algorithm

Initialise the distance matrix, where the direct travel cost between nodes is equivalent to the distance, and the travel cost from a node to the depot for refilling is double the travel distance due to logistics considerations

For each demand scenario iterate through all permutations of the nodes:

Compute the total travel cost according to TSP based on the distance matrix.

Now consider the demand values for that particular scenario and if the route includes visits to the depot for refilling, add the additional cost to the travel cost

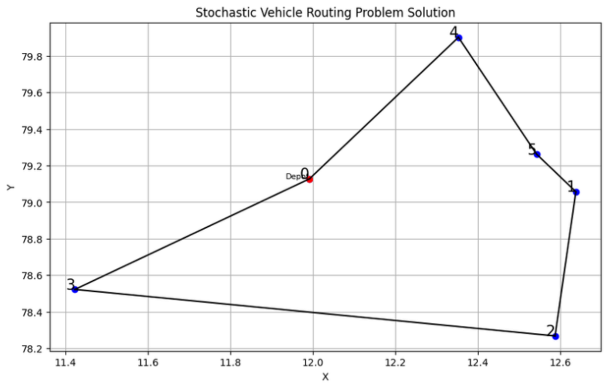
Define the minimization function as the sum of the travel cost and the probability of each scenario multiplied by the additional cost for that scenario. (express this in terms of variables)

Evaluate for all permutations of the nodes and determine the sequence of nodes that incurs the minimal total cost according to the minimization function.

Output the sequence of nodes representing the optimal route for minimal cost.

4.3.3 Limitations

1. The lesser number of modes makes it possible to check all permutations. Once the number of nodes increases, computation costs will increase, this is addressed in the



(a) Distance: 883, Route: 0-3-2-1-5-4-0

Figure 4.3: Distance: 883, Route: 0-3-2-1-5-4-0

next model.

2. Probabilities for scenarios are being considered here, this implies the individual node demands aren't independent, but in real world scenarios this may or may not be ideal. The next model considers individual node demands to solve the problem

4.4 Model 2.2: EV with node probabilities

4.4.1 Introduction

In this model every node is known to have a probability distribution function. For this report uniform and normal distributions are considered. This approach gives us more freedom to introduce stochasticity for each node with different parameters. For e.g. different standard deviations and means are allocated for the normal distribution assigned to all nodes

Variables

All variables are similar to the last model except that instead of demand scenarios and probabilities, we have distribution functions for each node. Thus we need to use this information to create demand scenarios and their probabilities before proceeding with the minimisation function. The algorithm to develop the same is given in the next section.

4.4.2 Algorithm

Creating demand scenarios based on individual node normal distributions:

1. Initialize parameters:
   (a) Define the demand distribution for each node using a normal distribution with different parameters.
   (b) Specify the number of demand scenarios to create (e.g., 200 scenarios).
   (c) E.g. Node 1: 'mean' : 40, 'std_dev': 5
2. Generate demand scenarios: For each node:

   (a) Sample demand values from the normal distribution to create demand scenarios.
   (b) Repeat this process for the desired number of scenarios
   (c) E.g. For Node 1: Demand = 40, probability = 0.5

3. Calculate probabilities for each demand scenario: For each demand scenario:
   (a) Calculate the probability of the scenario based on the sampled demand values for each node.
   (b) Multiply the probabilities of all nodes to obtain the joint probability for the scenario.
   (c) Store the joint probability for the scenario.
   (d) E.g. [40,50,30,25,35] then Joint Probability Distribution = $0.5\hat{5} = 0.03125$

4. Normalise probabilities:
   (a) Sum the probabilities of all scenarios and divide each scenario's probability by the sum to normalise the probabilities.
   (b) Ensure that the sum of all normalised probabilities equals 1.

5. Output:
   (a) The generated demand scenarios along with their corresponding normalised joint probabilities.

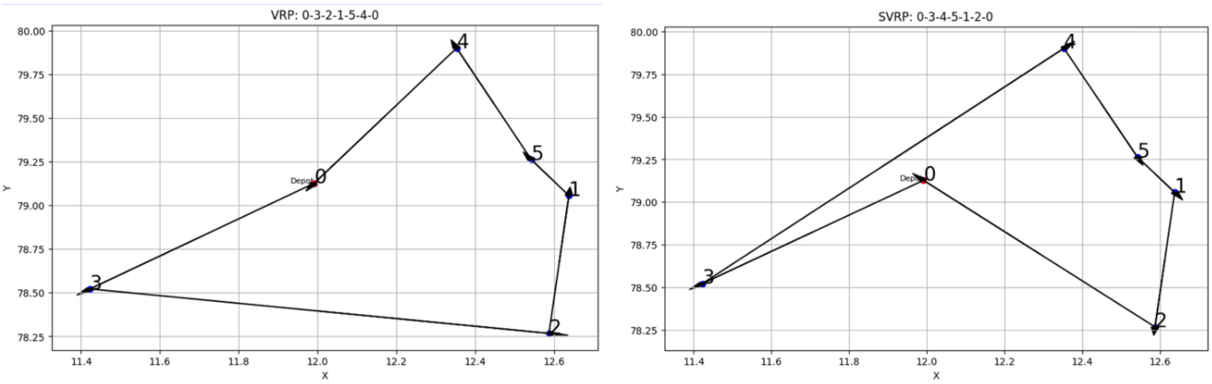
This algorithm outlines the systematic process of generating demand scenarios based on normal distributions for each node, computing the joint probabilities for these scenarios, and normalising the probabilities to ensure coherence. These steps facilitate the creation of probabilistic models for demand in vehicle routing optimization problems, enabling robust decision-making under uncertainty. These scenarios can then be implemented in the previous model 2.1 minimisation function to give the optimal sequence of nodes.

To address the limitation of permutation on a large number of nodes, Tabu Search algorithm is used to explore and refine the top 100 solutions for the Vehicle Routing Problem (VRP). Each solution is evaluated based on a minimization function fed into the Stochastic Vehicle Routing Problem (SVRP), which considers the uncertainties associated with customer demand. The performance of the SVRP-optimised solutions is then compared with those obtained from traditional VRP approaches. This comparative analysis sheds light on the efficacy of SVRP approach in generating robust solutions for complex routing problems, particularly in scenarios characterised by stochastic demand variations.

The performance of the algorithm across more nodes and varying vehicle capacity is depicted in plots below.

4.4.3 Conclusion

If we compare the node sequences we get using the SVRP and VRP calculated using mean demand values we can infer the following:

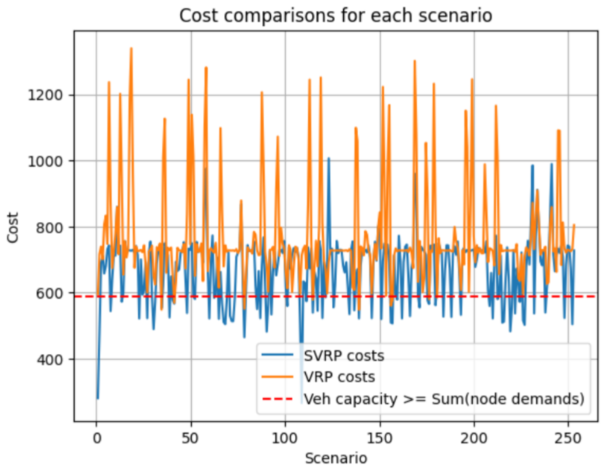


(a) VRP: 3-2-1-5-4-0 (b) SVRP: 0-3-4-5-1-2-0

Figure 4.4: SVRP Vs. VRP

1. SVRP solution outperforms the VRP for $\tilde{9}8\%$ of the scenarios
2. VRP solution outperforms SVRP for mere $\tilde{2}\%$ of the scenarios

This clearly defines the huge margin of cost savings achieved when the SVRP model is used. Also when tested on different datasets the supremacy of SVRP still persists.



(a) Distance comparisons across scenarios

Figure 4.5: Distance comparisons across scenarios

This approach is being extended to employee pick up given the following concerns:

1. **Uncertainty in travel times:** Stochastic VRP models can account for variability in travel times caused by factors like traffic congestion, road conditions, and unforeseen events, leading to more realistic and robust solutions.
2. **Dynamic customer requests:** Employee pick-up scenarios often involve dynamic requests throughout the day. Stochastic VRP models can handle these dynamic requests in real-time, allowing for adaptive route adjustments and efficient resource allocation.
3. **Workforce variability:** Uncertainty in workforce availability due to factors such as absenteeism or shift changes can be addressed in stochastic VRP models, ensuring better utilization of available resources.

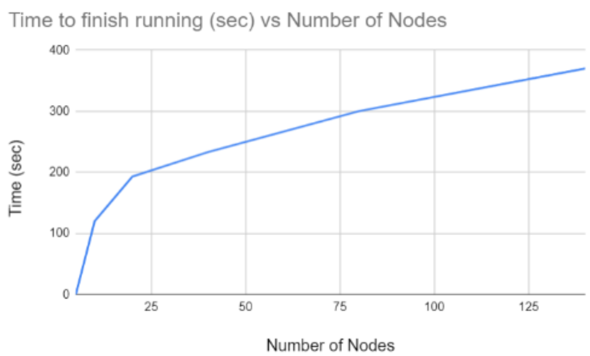
4. **Improved service quality:** By minimizing delays and reducing waiting times, stochastic VRP solutions enhance service quality for employee pick-ups, resulting in higher customer satisfaction.

5. **Cost optimization:** Stochastic VRP models optimize various cost components while considering uncertainties, leading to cost savings and improved resource utilization.

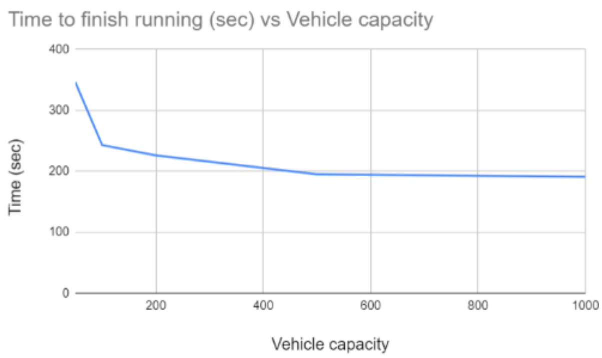
6. **Risk mitigation:** Explicitly considering uncertainties helps mitigate risks associated with disruptions or unforeseen events, enabling the development of contingency plans and backup routes.

7. **Scalability and flexibility:** Stochastic VRP models can scale to handle large-scale problems and adapt to incorporate additional constraints or objectives specific to employee pick-up operations, providing flexibility in addressing real-world complexities.

This can also be used to create a schedule for the vehicles and employees on daily, weekly or monthly basis depending on the extent of uncertainity a given industry demands. Eventually helping towards more operational efficiency and employee satisfaction.



(a) Time to finish running vs. number of nodes



(b) Time to finish running vs. no. of vehicles

Dataset No.	% Scenarios
Original Dataset	~95%
data_1	~97%
data_2	~92%
data_3	~96.5%
data_4	~97.3%
data_5	~96%

(c) % scenarios where SVRP outperforms VRP

Figure 4.6: SVRP model results

Chapter 5

Summary and Perspectives

In the final chapter, we summarize the key findings and perspectives derived from the study on vehicle routing problems (VRPs) with a focus on employee pick-up scenarios.

The research embarked on a systematic exploration of VRP models starting from single depot to multiple depots and eventually delving into the stochastic aspect in the context of addressing uncertainties inherent in travel times, dynamic customer requests, workforce variability, and other operational complexities. The investigation delved into the development and implementation of models on real data followed by model comparisons. The stochastic VRP was examined on a different problem statement and extended further to the original employee pick up problem.

Chapter 3 showcased the implementation of both the *Tabu search* and *genetic algorithm* models, aiming to demonstrate their effectiveness in outperforming existing Python libraries. Through comparisons between models, we explored how integrating and balancing the strengths of both algorithms could lead to optimal solutions, ultimately minimizing costs by achieving the best possible distance.

Chapter 4 delves deeper into enhancing the realism of modeling for Vehicle Routing Problems (VRP) by incorporating the crucial factor of stochasticity. Beginning with a focus on a smaller customer demand scenario, the chapter explores various approaches, with particular emphasis on the expected value method, which emerges as the most effective. This also helps us understand the importance of stochasticity for different industries. A comparison is then drawn with a conventional VRP formulation that considers only mean demands, highlighting the superiority of Stochastic VRP (SVRP) over VRP in approximately 95% of cases. This comparison underscores the efficacy of incorporating stochastic elements in VRP modeling to achieve more robust and reliable solutions.

In summary, a comprehensive analysis and investigation have been conducted to apply vehicle routing concepts to address the employee pick-up problem for an auto manufacturer based in Chennai. Given the vast scope of Vehicle Routing Problems (VRP) with numerous variations and diverse constraints, the question of how much to constrain the problem remains open-ended. However, replicating real-world scenarios with multiple constraints enhances problem complexity but also yields superior solutions. This is crucial for ensuring the efficient and satisfactory functioning of the organization, underscoring the importance of adapting VRP methodologies to address real-world challenges effectively.

Chapter 6

Future Scope

The research presented in this thesis has made significant strides in addressing the challenges of stochastic vehicle routing problems (SVRPs), particularly in the context of employee pick-up scenarios. However, the dynamic nature of real-world operations and the ever-evolving complexities of transportation and logistics necessitate further exploration and refinement.

One promising avenue for future work lies in the incorporation of real-time data and dynamic re-optimization capabilities. The current models assume static inputs and predefined scenarios, but the development of techniques to dynamically update the SVRP solutions based on real-time data streams, such as traffic conditions, unexpected events, or changes in customer requests, could significantly enhance their practical applicability. This would involve developing efficient algorithms for on-the-fly route adjustments and re-optimization, enabling the models to adapt to rapidly changing conditions seamlessly. Additionally, the integration of advanced optimization techniques from domains such as machine learning, deep learning, and reinforcement learning could potentially unlock new frontiers in model performance and adaptability. These techniques may provide more efficient solutions, particularly in large-scale or highly complex scenarios, further pushing the boundaries of what is achievable through traditional optimization algorithms.

In real-world scenarios, decision-makers often face the challenge of balancing multiple conflicting objectives, such as minimizing costs, maximizing customer satisfaction, reducing environmental impact, or optimizing resource utilization. Future research could explore multi-objective optimization techniques to generate Pareto-optimal solutions that strike an optimal balance between these competing objectives, providing decision-makers with a comprehensive understanding of the trade-offs involved. Moreover, the current models may not capture all the intricacies and complexities encountered in real-world employee pick-up operations. Future work could extend the models by incorporating additional constraints and operational complexities, such as time windows, heterogeneous vehicle fleets, driver availability, and shift scheduling, to enhance the models' practical applicability and ensure their relevance in diverse industrial settings.

To further validate the models' efficacy and facilitate their practical adoption, future work could involve collaborating with industry partners and conducting pilot studies or case studies in real-world employee pick-up scenarios. These collaborations could provide valuable insights, data, and feedback for refining and enhancing the models to better align with industry-specific requirements, ultimately promoting their widespread adoption. Finally, the development of user-friendly decision support systems or software interfaces that seamlessly integrate the models and their solutions could play a crucial role in promoting the widespread adoption of SVRP models in industry. These systems could provide intuitive visualizations, interactive interfaces, and decision-making support, enabling organizations to leverage the benefits of optimized routing solutions under stochastic demand conditions with ease.

By pursuing these avenues for future work, researchers and practitioners can further advance the field of stochastic vehicle routing problems, enhancing the models' robustness, adaptability, and practical applicability in the ever-evolving landscape of transportation and logistics operations, ultimately contributing to improved operational efficiency, cost savings, and enhanced customer satisfaction.

Chapter 7

Bibliography

1. Dimitris J. Bertsimas, "A Vehicle Routing Problem with Stochastic Demand," Massachusetts Institute of Technology, Cambridge, Massachusetts, (Received September 1988; revisions received February, November 1990; accepted May 1991)

2. Guoming Li and Junhua Li, "An Improved Tabu Search Algorithm for the Stochastic Vehicle Routing Problem With Soft Time Windows," Key Laboratory of Jiangxi Province for Image Processing and Pattern Recognition, Nanchang Hangkong University, Nanchang 330063, China.

3. Michel Gendreau, Gilbert Laporte, and René Séguin, "Stochastic vehicle routing," Invited Review, European Journal of Operational Research 88 (1996) 3-12.

4. Zangir Iklassov, Ikboljon Sobirov, Ruben Solozabal, and Martin Takáč, "Reinforcement Learning for Solving Stochastic Vehicle Routing Problem," MBZUAI, UAE, Abu-Dhabi.

5. Mohammadre za Nazari, Afshin Oroojlooy, Martin Takác, and Lawrence V. Snyder, "Reinforcement Learning for Solving the Vehicle Routing Problem," Lehigh University.

6. Zangir Iklassov, Ikboljon Sobirov, Ruben Solozabal, and Martin Takác, "Reinforcement Learning Approach to Stochastic Vehicle Routing Problem With Correlated Demands," Mohamed bin Zayed University of Artificial Intelligence (MBZUAI), Abu Dhabi, United Arab Emirates.

7. Nicola Secomandi and Francois Margot, "Reoptimization Approaches for the Vehicle-Routing Problem with Stochastic Demands," Operations Research, Vol. 57, No. 1, January-February 2009, pp. 214-230.

8. Ryota Omori and Takayuki Shiina, "Solution Algorithm for the Vehicle Routing Problem with Stochastic Demands," Department of Industrial and Management Systems Engineering, School of Creative Science and Engineering, Waseda University, Tokyo, Japan.

9. Yossiria Adulyasak and Patrick Jaillet, "Models and Algorithms for Stochastic and Robust Vehicle Routing with Deadlines," Singapore-MIT Alliance for Research and Technology (SMART), Singapore, and Department of Electrical Engineering and Computer Science, Operations Research Center, Massachusetts Institute of Technology, Cambridge, Massachusetts.

10. William R. Stewart, Jr. and Bruce L. Golden, "Stochastic vehicle routing: A comprehensive approach," published article, 1982.

11. Jorge Oyola, Halvard Arntzen, and David L. Woodruff, "The stochastic vehicle routing problem, a literature review, part I: models," published article in EURO Journal on Transportation and Logistics, 2016.

12. Youngbum Hur, Jonathan F. Bard, Markus Frey, and Ferdinand Kiermaier, "A stochastic optimization approach to shift scheduling with breaks adjustments," published article, 2019.

13. I.R. (Ian) Luik, "People Scheduling Service: an Employee Scheduling Algorithm based on Stochastic Workloads," Master's Thesis, Delft University of Technology, 2023.

14. Ming Liu, Bian Liang, Maoran Zhu, and Chengbin Chu, "Stochastic Check-in Employee Scheduling Problem," published article in IEEE Access, 2020.

15. Baker, B. M. and M. Ayechew. "A genetic algorithm for the vehicle routing problem." Computers & Operations Research, 30(5), pp. 787–800, 2003.

16. Baker, B. M. and M. Ayechew. "A genetic algorithm for the vehicle routing problem." Computers & Operations Research, 30(5), pp. 787–800, 2003.

17. Beasley, J. E. "Route first—cluster second methods for vehicle routing." Omega, 11(4), pp. 403–408, 1983.

18. Berger, J. and M. Barkaoui. "A hybrid genetic algorithm for the capacitated vehicle routing problem." In Genetic and evolutionary computation conference. Springer, 2003.

19. BoussaïD, I., J. Lepagnot, and P. Siarry. "A survey on optimization metaheuristics." Information sciences, 237, pp. 82–117, 2013.

20. Brandão, J. "A lower bound based meta-heuristic for the vehicle routing problem." In Essays and surveys in metaheuristics, pp. 151–168. Springer, 2002.

21. Brandão, J. "A tabu search algorithm for the open vehicle routing problem." European Journal of Operational Research, 157(3), pp. 552–564, 2004.

22. Clarke, G. and J. W. Wright. "Scheduling of vehicles from a central depot to a number of delivery points." Operations research, 12(4), pp. 568–581, 1964.

23. Cordeau, J.-F. and M. Maischberger. "A parallel iterated tabu search heuristic for vehicle routing problems." Computers & Operations Research, 39(9), pp. 2033–2050, 2012.

24. da Costa, P. R. d. O., S. Mauceri, P. Carroll, and F. Pallonetto. "A genetic algorithm for a green vehicle routing problem." Electronic notes in discrete mathematics, 64, pp. 65–74, 2018.

25. Dantzig, G., R. Fulkerson, and S. Johnson. "Solution of a large-scale traveling salesman problem." Journal of the operations research society of America, 2(4), pp. 393–410, 1954.

26. Dantzig, G. B. and J. H. Ramser. "The truck dispatching problem." Management science, 6(1), pp. 80–91, 1959.

27. Deif, I. and L. Bodin. "Extension of the clarke and wright algorithm for solving the vehicle routing problem with backhauling." In Proceedings of the Babson conference on software uses in transportation and logistics management. Babson Park, MA, 1984.

28. Dijkstra, E. W. "A note on two problems in connexion with graphs." Numerische mathematik, 1(1), pp. 269–271, 1959.

29. Eastman, W. "Linear programming with pattern constraints." Ph.D. thesis, Department of Economics, Harvard University, Cambridge, Massachusetts, USA, 1958.

30. Eilon, S., C. Watson-Gandy, and N. Christofides. "Distributed management." Hafner, New York, 1971.

31. Fisher, M. L. and R. Jaikumar. "A generalized assignment heuristic for vehicle routing." Networks, 11(2), pp. 109–124, 1981.

32. Fu, Z., R. Eglese, and L. Y. Li. "A new tabu search heuristic for the open vehicle routing problem." Journal of the Operational Research Society, 56(3), pp. 267–274, 2005.

33. Fu, Z. and M. Wright. "Train plan model for British rail freight services through the channel tunnel." Journal of the Operational Research Society, 45(4), pp. 384–391, 1994.

34. Gaskell, T. "Bases for vehicle fleet scheduling." Journal of the Operational Research Society, 18(3), pp. 281–295, 1967.

35. Gendreau, M., A. Hertz, and G. Laporte. "A tabu search heuristic for the vehicle routing problem." Management science, 40(10), pp. 1276–1290, 1994.

36. Gillet, B. and L. Miller. "A heuristic algorithm for the vehicle dispatch problem." Operational research, 1974.

37. Glover, F. "Future paths for integer programming and links to artificial intelligence." Computers & operations research, 13(5), pp. 533–549, 1986.

38. Goetschalckx, M. and C. Jacobs-Blecha. "The vehicle routing problem with backhauls." European Journal of Operational Research, 42(1), pp. 39–51, 1989.

39. Golden, B. L., T. L. Magnanti, and H. Q. Nguyen. "Implementing vehicle routing algorithms." Networks, 7(2), pp. 113–148, 1977.

40. Jozefowiez, N., F. Semet, and E.-G. Talbi. "Multi-objective vehicle routing problems." European journal of operational research, 189(2), pp. 293–309, 2008.

41. Laporte, G. "The vehicle routing problem: An overview of exact and approximate algorithms." European journal of operational research, 59(3), pp. 345–358, 1992.

42. Li, L. and Z. Fu. "The school bus routing problem: a case study." Journal of the Operational Research Society, 53(5), pp. 552–558, 2002.

43. Lin, S. and B. W. Kernighan. "An effective heuristic algorithm for the traveling salesman problem." Operations research, 21(2), pp. 498–516, 1973.

44. Miller, C. E., A. W. Tucker, and R. A. Zemlin. "Integer programming formulation of traveling salesman problems." Journal of the ACM (JACM), 7(4), pp. 326–329, 1960.

45. Min, H., V. Jayaraman, and R. Srivastava. "Combined location-routing problems: A synthesis and future research directions." European Journal of Operational Research, 108(1), pp. 1–15, 1998.

46. Newton, R. M. and W. H. Thomas. "Design of school bus routes by computer." Socio-Economic Planning Sciences, 3(1), pp. 75–85, 1969.

47. Nurcahyo, G. W., R. A. Alias, S. M. Shamsuddin, and M. N. M. Sap. "Sweep algorithm in vehicle routing problem for public transport." Jurnal Antarabangsa Teknologi Maklumat, 2, pp. 51–64, 2002.

48. Örmeci, E. L., F. S. Salman, and E. Yücel. "Staff rostering in call centers providing employee transportation." Omega, 43, pp. 41–53, 2014.

49. Osman, I. H. "Metastrategy simulated annealing and tabu search algorithms for the vehicle routing problem." Annals of operations research, 41(4), pp. 421–451, 1993.

50. Osman, I. H. and G. Laporte. "Metaheuristics: A bibliography."

51. Pitakaso, R., K. Sethanan, and N. Srijaroon. "Modified differential evolution algorithms for multi-vehicle allocation and route optimization for employee transportation." Engineering Optimization, 52(7), pp. 1225–1243, 2020.

52. Saeheaw, T. and N. Charoenchai. "Integration of geographical information systems, meta-heuristics and optimization models for the employee transportation problem." Journal of Spatial Science, 62(2), pp. 281–306, 2017.

53. Sahu, H. et al. "Optimized solution for employee transportation problem using linear programming." In Smart Innovations in Communication and Computational Sciences, pp. 247–255. Springer, 2019.

54. Sariklis, D. and S. Powell. "A heuristic method for the open vehicle routing problem." Journal of the Operational Research Society, 51(5), pp. 564–573, 2000.

55. Schrage, L. "Formulation and structure of more complex/realistic routing and scheduling problems." Networks, 11(2), pp. 229–232, 1981.

56. Solomon, M. M. "Algorithms for the vehicle routing and scheduling problems with time window constraints." Operations research, 35(2), pp. 254–265, 1987.

57. Tarantilis, C. D. and C. T. Kiranoudis. "Boneroute: An adaptive memory-based method for effective fleet management." Annals of operations Research, 115(1-4), pp. 227–241, 2002.

58. Tyagi, M. "A practical method for the truck dispatching problem." Journal of the Operations Research Society of Japan, 10, pp. 76–92, 1968.

59. Yellow, P. "A computational modification to the savings method of vehicle scheduling." Journal of the Operational Research Society, 21(2), pp. 281–283, 1970.

60. Zhao, Y., H. Zhou, and Y. Liu. "A Cost-Effective Offline Routing Optimization Approach to Employee Shuttle Services." Technical report, SAE Technical Paper, 2017.