University of South-Eastern Norway

www.usn.no

FMH606 Master's Thesis 2022
Electrical Power Engineering

OPTIMAL SCHEDULING FOR MOBILE BATTERY CHARGING SYSTEMS

Dhanush Wagle

Faculty of Technology, Natural Sciences and Maritime Sciences
Campus Porsgrunn

University of South-Eastern Norway

www.usn.no

Course: FMH606 Master's Thesis, 2022

Title: Optimal scheduling for mobile battery charging systems

Number of pages: 53

Keywords: Routing, Scheduling, Vehicle Scheduling Problem, Optimization

Student: Dhanush Wagle

Supervisor: Carlos F. Pfeiffer

External partner: Skagerak Energi

Summary:

A pilot project in Norway is currently being examined to give electric energy to construction sites in places where connection to the power grid is not available. This thesis proposes a generic mobile battery charging scheduling problem that entails charging batteries in a location where the grid has adequate capacity and then moving the batteries from the charging station to relevant construction sites that use battery-powered construction machines. The fundamental principle behind vehicle routing problems with a time window and battery electric transit vehicle scheduling problem has been useful for problem formulation. To address these formulations, mixed-integer linear programming, and large neighborhood search algorithms are being investigated. The optimization model is formulated as mixed-integer linear programming, with objective functions, constraints, and other important parameters, and then solved with the Microsoft Excel solver using a large neighborhood search algorithm. Two study cases are formulated: a simple optimization problem to help understand the notion of vehicle routing problem and a more complex scheduling problem based on a real-world scenario.

Preface

This report contributes to the final work for course FMH606 Master`s Thesis of Master of Science Electric Power Engineering, at the Department of Electrical Engineering, IT and Cybernetic at the University of South-Eastern Norway (USN).

This thesis aims to research and analyze the optimal scheduling problem considering several mobile battery containers, several possible charging stations, and several construction sites with different needs.

I would like to thank my supervisor, Professor Carlos F. Pfeiffer, for the supervision, expert guidance, and helpful insights during the work on this thesis and my co-supervisor Associate Professor Thomas Øyvang for his instructions and collaboration in my thesis.

I would also like to take the chance to express my appreciation to USN, who gave me such a precious chance to study in Norway, and to express my thanks to Kenneth Anderson, Håvard Hadland as well as whole Skagerak Energi team, the external partner for presenting relevant insights and data required to complete this thesis.

May 18, 2022

Porsgrunn

Dhanush Wagle

Contents

List of Abbreviations

ALNS	–	Adaptive Large Neighborhood Search
BET-VSP	–	Battery-Electric Transit Vehicle Scheduling Problem
DH	–	Deadheading
EV	–	Electrical Vehicle
E-VSP	–	Electric Vehicle Scheduling Problem
ft	–	Feet
GA	–	Genetic Algorithm
kW	–	Kilo Watt
kWh	–	Kilo Watt Hour
LIFO	–	Last-In First-Out
LNS	–	Large Neighborhood Search
LP	–	Linear Programming
MD-VSP	–	Multi-Depot Vehicle Scheduling Problem
MILP	–	Mixed Integer Linear Programming
MINLP	–	Mixed Integer Non-Linear Programming
MIP	–	Mixed-Integer Programming
MW	–	Mega Watt
NLP	–	Non-Linear Programming
TSP	–	Travelling Salesman Problem
VRP	–	Vehicle Routing Problem
VRPTW	–	Vehicle Scheduling Problem with Time Window
VSP	–	Vehicle Scheduling Problem

List of Figures

List of Tables

1 Introduction

1.1 Background

A fossil-free construction site is one that does not use fossil fuel-powered machines, but an emission-free construction site is one that uses construction technologies such as a battery, cable-electric, or hydrogen to produce zero emissions. The fossil-free construction site typically employs bio-fueled construction equipment that are carbon neutral but does produce other pollutants such as particulate matter and nitrogen oxide, implying that fossil-free does not imply zero emissions. Even while electric construction machines can be utilized in both fossil-free and emission-free building sites, their usage in zero-emission sites is more likely to make the site emission-free.

A pilot project to provide electric energy to construction activities in parts where access to the power grid is not possible is presently under evaluation in Norway. This project focuses on zero-emission construction sites of Skagerak Energi. A lot of places have limited possibilities for connection in the electric grid and extending the grid just for the construction activities is also not recommended. Mobile energy storage might resolve the issue related to grid connection and power up electric construction machines at sites. As described in Figure 1-1, the idea is to use mobile battery containers to charge batteries at a location where the grid has good capacity and then drive the batteries from the charging station to the relevant construction sites that use battery-powered construction machines. When the batteries are discharged, the empty batteries are driven to the nearest charging station, and new fully charged batteries are driven to the construction site.

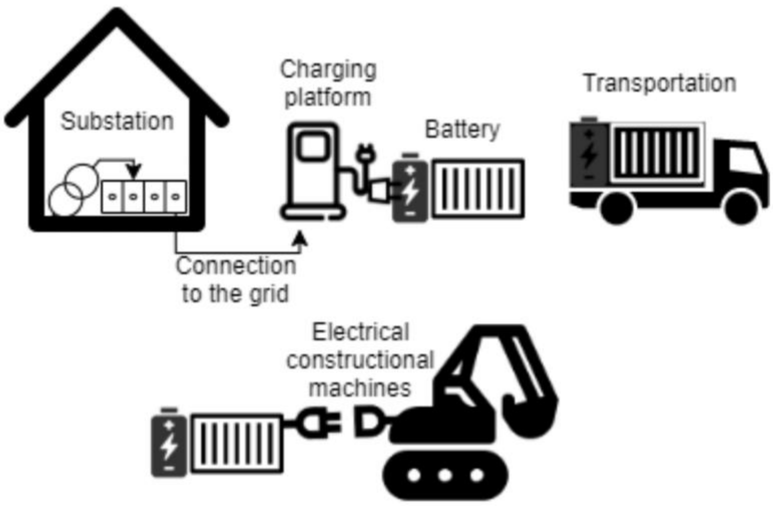

Figure 1-1: An overview of battery container from construction site to grid connection [1]

At an increasing rate, new technologies are emerging and becoming a part of our daily lives. The smartphone revolution, as well as the evolution of automobiles and new energy technologies, have resulted in a technological break from traditional procedures and thinking. This has an impact on all aspects of daily life, including the planning and operation of transportation. Electric vehicles are increasingly being used by public-transit agencies due to their lower emissions and other social and economic benefits.

1.2 Previous Skagerak Work

This thesis stands on the foundation laid by the previous work carried out by a group of students as a master's thesis at USN in collaboration with Skagerak Energi. The findings of the project are used as a base case for this thesis for the optimal scheduling of mobile battery charging systems. The findings useful for this thesis are listed below [1]:

1. The most prevalent size of the mobile battery used in Norway is a one with 7.5-ton weight and 576 kWh energy. The standard maximum dimension of the container carrying the mobile battery is 45ft long, 8 ft wide and 8 ft 6 inches tall.
2. A typical construction site (10,000 square meters apartment block or school) needs three excavators (250 kWh each) and one mobile crane (500 kWh) and diverse small machines (150 kWh) energy. The construction site needs a total of 1400 kWh energy per day i.e., three 576 kWh batteries.
3. The combined charging system (CCS) type 2 or Combo 2 cable can be used in the charging station as it can provide power at up to 350 kW. The charging time for one battery using CCS type 2 is 2 hours.
4. The three possible charging stations are:
    a. Hauen – ideal connection point – 8.8 MW max loading capacity – can charge max 15 batteries
    b. Tømmerkaia – strained grid – 4.1 MW max loading capacity – can charge max 7 batteries
    c. Floodmyrvegen – recommended by Skagerak

1.3 Objectives

- Propose a general mobile battery charging scheduling problem
- Determine important variables and constraints
- Propose a realistic case scenario for the scheduling problem
- Formulate an optimization model, with objective functions, constraints, and other important parameters
- Solve the optimization problem using adequate tools and algorithms

1.4 Limitations

- Only fixed cost and cost per unit distance for vehicles, remaining cost variables like chargers, stations, etc. are neglected.
- Fuel consumption and CO_2 emission are not regarded because of the complexity of the problem.
- Cases that require the vehicle to go to the same place more than once in a given schedule date, are not considered.
- Could have been more polished to get better results with a few more added features like battery partial charging/discharging and emissions if thesis time was not limited
- If one or more constraint is not fulfilled, the algorithm is infeasible to show an alternative solution.

2 Vehicle Routing and Scheduling

The following sub-section highlight's themes and ideas that form the basis of the scheduling problem.

2.1 Traveling Salesman Problem

The traveling salesman problem (TSP) is one of the well-known problems in optimization, logistics, or operations research. It is a task where a salesman must visit a list of pre-defined cities once and return to the city, he started from in the shortest route possible. It is an NP-hard combinatorial optimization problem that is important in theoretical computer science and operations research. It is one of management science's most studied problems. Mathematical programming is used to solve optimal approaches to traveling salesman problems. However, most TSP problems are not solved optimally. Heuristics are used when the problem is so large that an optimal solution is impossible to obtain, or when approximate solutions are sufficient. The Clark and Wright savings heuristic and the nearest neighbor technique are two regularly utilized TSP heuristics. In 1972, Karp showed that TSP is an NP-hard which was one of the first problems to be shown NP-hard while the notion of NP-completeness was still developing. New algorithmic strategies have first been created for or at least applied to, the TSP to demonstrate their efficacy. Branch and bound, Lagrangian relaxation, Lin-Kernighan type approaches, simulated annealing, and polyhedral combinatorics for hard combinatorial optimization problems are some examples (polyhedral cutting plane methods and branch and cut). [2]

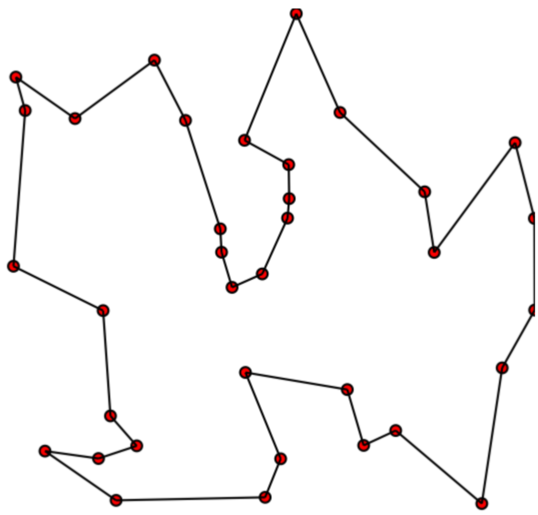

Figure 2-1: TSP Solution - Shortest Possible Loop [3]

TSP can be formulated as minimizing a function

$$\sum_{i=1}^{n} c_{i\pi(i)}$$

Where π is cyclic permutation of the integers from 1 to n given an integer $n \geq 3$ and an $n \times n$ matrix $C = (c_{ij})$, each c_{ij} is a non-negative integer.

The TSP has been first documented around the mid 1700s so, is a relatively old problem. Euler was interested in solving the knights' tour problem which was basically a problem a knight should visit each of the 64 squares of a chessboard exactly once on its tour, The phrase 'traveling salesman' first appeared in a German book authored by a senior traveling salesperson in 1932. The phrase "traveling salesman dilemma" was first used in a publication by the RAND Corporation in 1949. The Corporation's reputation contributed to the TSP being a well-known and popular problem. The TSP gained popularity at the same time as a result of the new subject of linear programming and attempts to tackle combinatorial issues. [4][5]

2.2 Vehicle Routing Problem

The classic vehicle routing problem (VRP) extends the numerous traveling salesman problem by incorporating varied service requirements at each node as well as different vehicle capacities in the fleet to reduce the overall cost or distance along all routes. The main goal of VRP is used to find the optimal set of routes for vehicles delivering goods to clients to minimize the total route cost. VRP is an NP-hard problem therefore commercial or practical problem solvers prefer the heuristics approach due to the size and frequency of real-world scenarios. It first appeared in 1959 as 'The Truck Dispatching Problem' by George Dantzig. [6] [7]

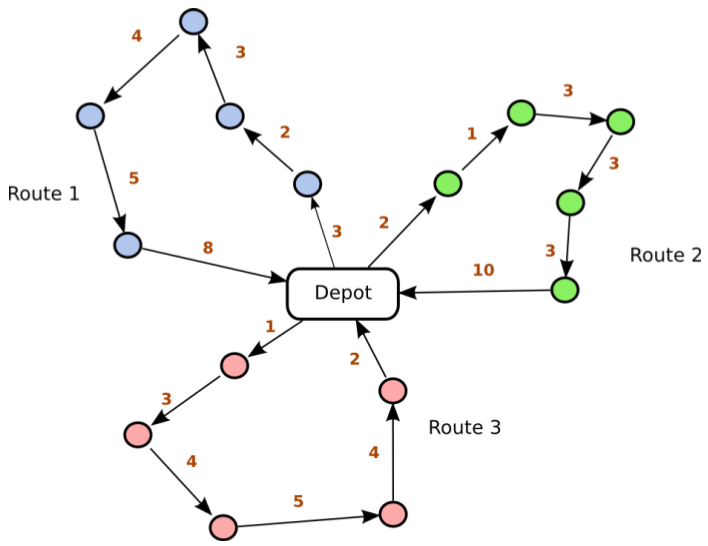

Figure 2-2: Vehicle Routing Problem [6]

Dantzig, Fulkerson and Johnson extended a TSP to create the two-index vehicle flow to minimize the total cost of the route: [6]

$$min \sum_{i \in V} \sum_{j \in V} c_{ij} x_{ij}$$

Where c_{ij} and x_{ij} represent the total cost and binary variable that represent the part of the solution while traveling from point i to j. The constraints for this minimization functions were formulated as:

- Constraints that state exactly one arc enters and leaves each vertex associated with a customer
- Constraints ensure that the number of vehicles leaving and entering the depot is the same.

- Capacity cut constraints, which impose that the routes must be interconnected, and the demand on each route must not exceed the capacity of the vehicles.
- Integrality constraints

There are several variations of VRP, some of which are:

1. Vehicle Routing Problem with Profits
2. Vehicle Routing Problem with Pickup and Delivery
3. Vehicle Routing Problem with LIFO
4. Vehicle Routing Problem with Time Windows
5. Capacitated Vehicle Routing Problem
6. Vehicle Routing Problem with Multiple Trips
7. Open Vehicle Routing Problem
8. Inventory Routing Problem
9. Multi-Depot Vehicle Routing Problem

2.3 Vehicle Routing Problem with Time Window

The vehicle routing problem with time window (VRPTW) is an important concept in logistics systems that has received a lot of attention in recent years where many researchers have contributed to formulating and solving the optimization problem. The routing problem with a goal to minimize the total transportation cost can be stated as selecting routes for a limited number of cars to serve a group of consumers within time constraints where each vehicle has a maximum carrying capacity that begins and ends at the depot and each customer is served only once. Schrage identified the vehicle routing and scheduling problem with time window limitations as an important area for advancement in dealing with realistic complexities and generalizations of the fundamental routing model in 1981. [8]

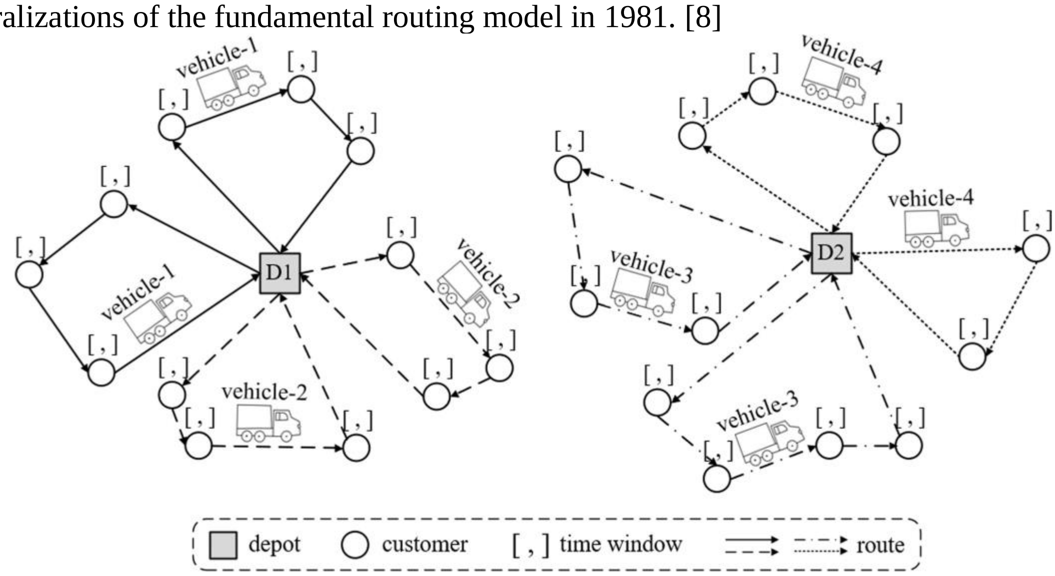

Figure 2-3: Multi-Depot VRPTW [9]

Without time windows, VRP is NP-complete. Even with a fixed fleet of vehicles, the time-constrained problem is fundamentally more difficult than a simple VRP. It is NP-complete to find a feasible solution for a VRPTW with fixed fleet size. Because of the inherent difficulty of the problem, heuristic-based search methods are most promising for solving practical size problems. In a reasonable amount of computer time, heuristic methods frequently produce optimal or near-optimal solutions for large problems. As a result, the development of heuristic algorithms capable of obtaining near-optimal feasible solutions for large VRPTW is of primary importance. [10]

2.4 Vehicle Scheduling Problem

Most routing and scheduling problems seek to reduce the total cost of providing the service. which includes the total cost or price of the vehicle, mileage, and personnel costs. However, other goals, particularly in the public sector, may come into play. Routing and scheduling issues are frequently represented as graphical networks as shown in Figure 2-4 which has the benefit of allowing the decision-maker to see the problem in context. The classification of routing and scheduling problems is determined by the characteristics of the service delivery system, such as the size of the delivery fleet, the location of the fleet, vehicle capacities, and routing and scheduling objectives. We begin with a collection of nodes that will be visited by a single car in the most basic case. Transit costs between nodes are the same regardless of direction, there are no precedence relationships between them moreover there are no delivery-time constraints. Furthermore, the vehicle capacity is not considered. The solution to the single-vehicle dilemma is a route or tour that visits each node only once and starts and ends at the depot node. The tour is designed with the purpose of lowering the overall tour cost in mind. The simplest case is known as the traveling salesman problem (TSP). When we restrict the capacity of the various vehicles and add the possibility of fluctuating demands at each node, the problem is classed as a vehicle routing problem (VRP). If the customers being served have no time constraints and no precedence relationships exist, the problem is purely routing. If the service must be performed at a specific time, there is a scheduling issue. Otherwise, it's a routing and scheduling problem. [7]

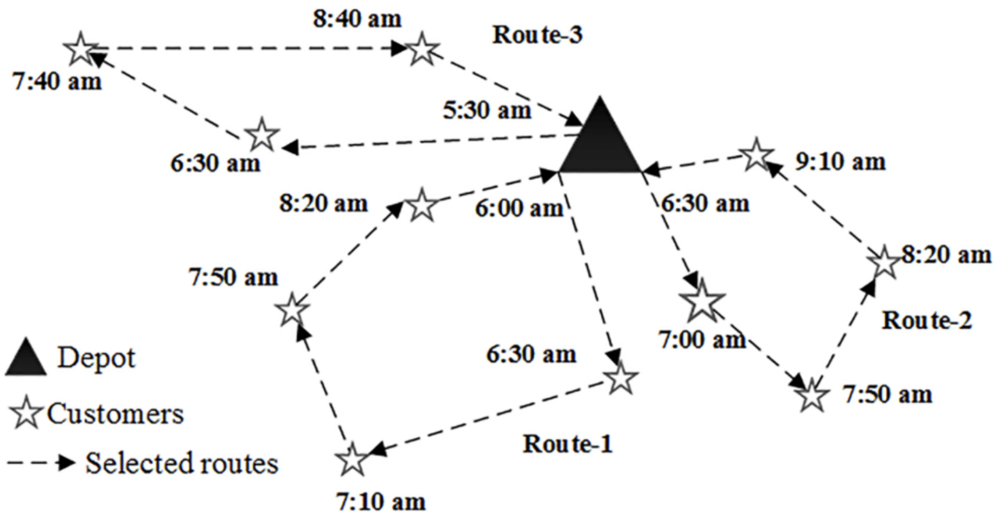

Figure 2-4: Visualization of Vehicle Scheduling Problem [11]

The problem of determining the optimal allocation of vehicles to carry out all the trips in each transit timetable is referred to as vehicle scheduling. Each vehicle is assigned a series of trips, which may include deadheading or empty trips. The number of viable solutions to this problem is extremely large, especially when the vehicles are in multiple depots. It is a classical non-linear programming problem in the field of modern operational research and logistics management. VSP can be subdivided into static and dynamic. Dynamic VSP refers to the optimization in the process of logistics and distribution to find the best route based on the new service request. [12]

2.5 Electric Vehicle Scheduling Problem

The electric vehicle scheduling problem (E-VSP) is a variant of the VSP that considers the restricted driving range and charging requirements of electric vehicles. The light-duty private car and light truck sector have led the way in the adoption of hybrid and battery electric technology during the last 20 years, while electric buses in the medium to a heavy-duty category are also gaining popularity. Although electric buses have numerous advantages, range anxiety is a prominent worry among electric bus users. In general, most diesel buses have a maximum driving range of more than 300 kilometers in urban conditions; however, the maximum driving range of most electric buses currently on the market ranges from 70 to 200 kilometers, which is 25–65 percent less than that of diesel buses, making it difficult to operate them continuously without recharging. [13]

The energy consumed must be immediately supplied using either battery switching or fast charging technology to assure proper operations. It must schedule battery changes for electric buses, identify the minimum quantity of spare batteries to stock, and schedule recharging for the spare batteries for the battery swapping mode, all of which are areas where significant research progress has already been made. It must schedule the charging of electric buses, establish the placement of charging stations, and determine the number of chargers required for the fast-charging mode. It is critical to design a cost-effective decision-making framework that can provide optimal strategies for planning and operational decisions while meeting the recharging demand of all-electric buses without delays or congestion to make the most of a fast-recharging system. Comprehensive planning decisions about the location and capacity of charging stations are used to make operational decisions about the recharging schedule or the assignment of electric buses to chargers. To achieve overall cost-effectiveness, both planning and operational choices must be made concurrently in an integrated modeling framework [14]. In the literature, the Vehicle Scheduling Problem (VSP) has been thoroughly investigated and expanded to other versions, including the Multi-Depot VSP (MD-VSP), the Multiple Vehicle Types VSP, and the VSP with Route Constraints (VSP-RC) where various forms of route constraints, such as route time, route distance, or maximum vehicle bus line modifications, can be imposed [7].

The E-VSP is an MDVSP with distance limits and charging capabilities. Each trip in the E-VSP begins and ends at certain locations at predetermined times. Each vehicle can be fully or partially recharged at any recharging station. The charging time is believed to be a linear function of the battery's charge level. An E-VSP solution is a collection of vehicle schedules in which each vehicle begins, and each route is covered by precisely one vehicle, and it all starts and concludes at its base depot and the driving ranges of the vehicles are not exceeded. The goal is to first reduce the number of cars used and then to reduce the overall distance traveled. Because the traveling distance of each trip is fixed, minimizing the distance between the depot and the trip is analogous to reducing overall travel distance, as well as the distance between any two excursions in the schedule, also known as deadheading distance [15].

2.6 Battery Electric Transit Vehicle Scheduling Problem

Electric vehicles are being deployed in an increasing number of transit agencies throughout the world due to zero emissions and other social and economic benefits. One of the most difficult jobs is successfully arranging a group of EVs while keeping in mind the restricted driving range and charging requirements. This results in the battery-electric transit vehicle scheduling problem (BET-VSP), which is exacerbated by stationary battery chargers provided at transit terminals. From both a theoretical and practical standpoint, the BET-VSP is a novel and crucial research challenge. The BET-VSP takes into account stationary chargers deployed at transit terminals with the goal of minimizing not only the total number of electric transit vehicles required but also the total number of battery charges required to conduct a particular set of planned services. [12]

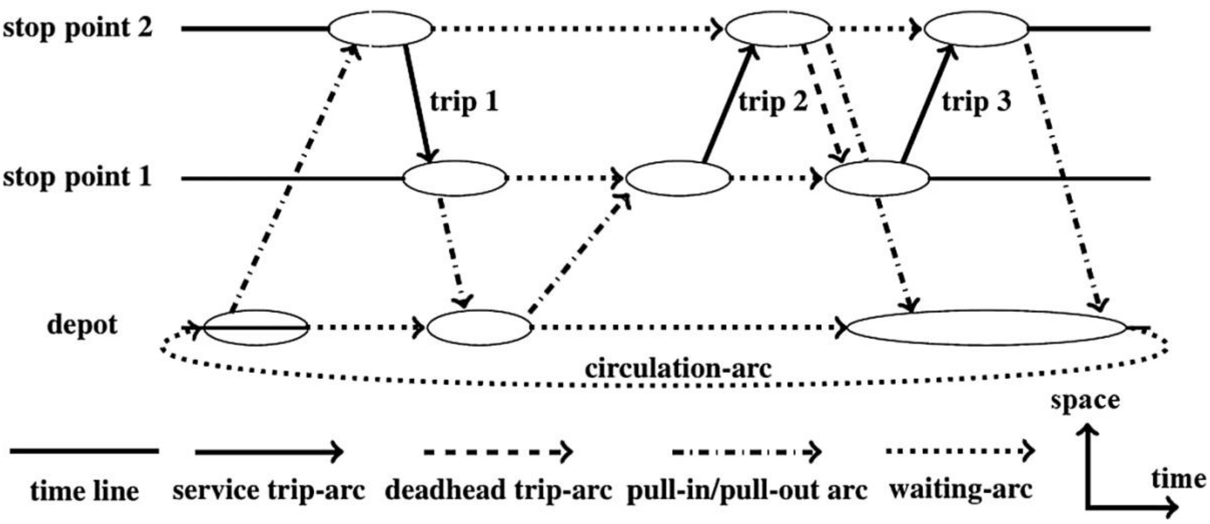

Figure 2-5: BET-VSP Example with space-time network [16]

Researchers Sassi and Oulamara investigated the single-depot electric vehicle scheduling problem, developed an EVSP mixed-integer programming model, and evaluated the cooperative scheduling and charging of electric vehicles. The topic seeks to maximize vehicle assignment to tours and reduce EV charging costs while satisfying operational limits on charging stations, the power grid, and EV driving range. [17]

MD-EVSP is more complex and difficult to solve than the single-depot electric vehicle scheduling problem (SD-EVSP). To date, there is just the following literature on this topic:
To overcome the problem Wen et al [15] developed a mixed-integer programming model for MD-EVSP and offered an adaptive big neighbor-hood search algorithm. For big instances of MD-EVSP, this technique can create decent solutions, while for small instances, it can generate optimal or near-optimal solutions.

Li et al. [18] proposed a formulation for the multi-depot vehicle scheduling problem with different vehicle types, including electric buses, under range and refueling constraints. A simpler formulation was created to generate a feasible spatiotemporal energy network for bus traffic and for passenger flow.

3 Problem Description

For a general formulation of the problem, a scenario in which a construction company intends to replace all conventional diesel equipment with electric counterparts in an emission-free construction site is investigated, and sensitivity analyses are performed under various electrification rate scenarios. Because the operational ranges of electric equipment are less than those of diesel counterparts, many charging stations equipped with fast chargers will be strategically installed at the charging stations to assure regular functioning. Given the time constraints, the following decisions are to be made.

- Total number of charging stations required
- Total number of chargers in each charging station
- Total number of mobile battery containers to be scheduled so that they may be recharged without any delays or charging station congestion

An optimization framework is created in this section. Because the suggested architecture is generic, it can be applied to a variety of networks.

3.1 Problem Definition

The main tasks in this study are to schedule the mobile battery container to be recharged and to determine the number of the charging stations and needed chargers. Several major assumptions are made to simplify the problem.

1. The mobile battery container operates according to the previously established timetable.
2. All the mobile battery containers are homogeneous and have the same driving range.
3. The charge consumed is proportional to the working hours on the construction site.
4. The time it takes to charge is directly proportional to the amount of energy recharged.
5. All chargers are homogeneous fast chargers with one outlet each.
6. The recharging duration is fixed, and continuous.

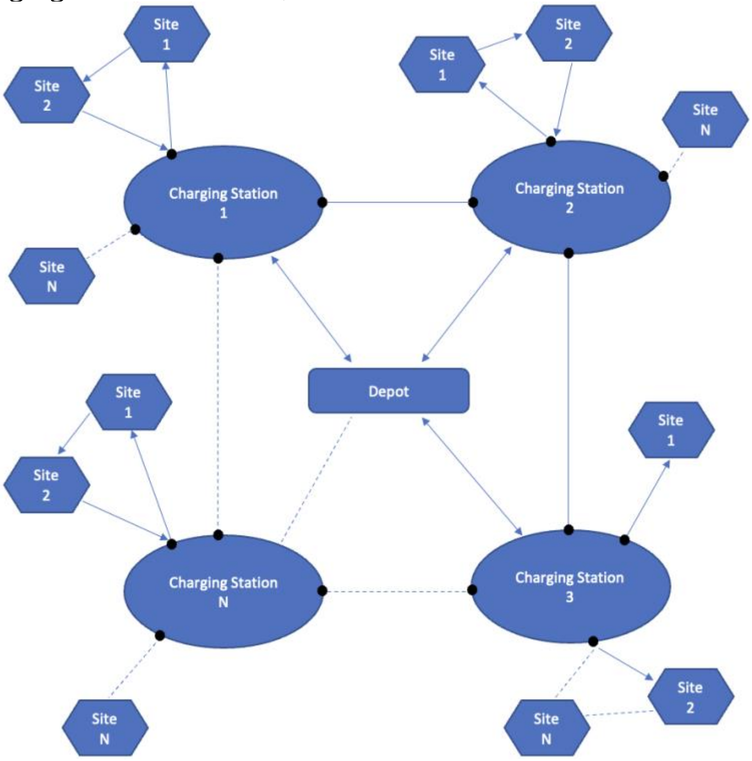

Figure 3-1: Single depot charging network architecture

3.2 Mathematical Formulation

The mathematical formulation used in this chapter is introduced in [14].

Minimize:

$$C = \sum_{i\in S}\sum_{n\in N}(c_d d_{in} + c_0 + c_e u_n)\bar{D}X_i^n + \sum_{k=1}^{K_n}\sum_{n\in N}\sum_{(i,j,t)\in A_n} c_w w_{ijnt}\bar{D}Y_{it}^{nk} + \sum_{k=1}^{K_n}\sum_{n\in N}\left(\alpha c_{f1} + \bar{D}c_{m1}\right)Z_{nk} + \sum_{n\in N}\left(\alpha c_{f2} + \bar{D}c_{m2}\right)Z'_k \quad (1)$$

Subject to:

$$\sum_{n\in N} X_i^n \leq 1, \forall\, i \in S \quad (2)$$

$$E_i + \sum_{n\in N}(\theta u_n - d_{in})X_i^n \leq \beta, \forall\, i \in S \quad (3)$$

$$E_j = E_i + \sum_{n\in N}(\theta u_n - d_{in})X_i^n - d_j, \forall\, (i,j) \in P \quad (4)$$

$$E_i \geq e_{min}, \forall\, i \in S \quad (5)$$

$$\sum_{k=1}^{K_n}\sum_{(i,j,t)\in A_n} Y_{it}^{nk} = X_i^n, \forall\, i \in S, n \in N \quad (6)$$

$$\sum_{i:(i,j,t')\in A_n}\sum_{t'=t-u_n+1}^{t} Y_{it'}^{nk} \leq 1, \forall\, k = 1, \dots, K_n, n \in N t \in T \quad (7)$$

$$\sum_{(i,j,t)\in A_n} Y_{it}^{nk} \leq \bar{M}Z_{nk}, \forall\, k = 1, \dots, K_n, n \in N \quad (8)$$

$$\sum_{k}^{K_n} Z_{nk} \leq \bar{M}Z'_n, \forall\, n \in N \quad (9)$$

$$Z_{nk} \leq Z_{n(k-1)}, \forall\, n \in N, k = 2, \dots, K_n \quad (10)$$

$$X_i^n = \{0,1\}, \forall\, i \in S, n \in N \quad (11)$$

$$Y_{it}^{nk} = \{0,1\}, \forall\, i \in S, t \in T, k = 1, \dots, K_n, n \in N \quad (12)$$

$$Z_{nk} = \{0,1\}, \forall\, k = 1, \dots, K_n, n \in N \quad (13)$$

$$Z'_n = \{0,1\}, \forall\, n \in N \quad (14)$$

$$E_i = \beta_0, \forall\, i \in O \quad (15)$$

Parameters

β – maximum amount of energy for a fully charged mobile battery container, in kWh
β_0 – the initial amount of energy for a mobile battery container at the depot, in kWh
θ – recharging rate, i.e., the extended charge using energy charged per hour, in kW
d_i – energy used in a day $i \in S$, in kWh
d_{in} – energy used between the start/end point of trip $i \in S$ and charging station n ∈ N, kWh
c_w – cost of unit waiting time, in NOK/hour
e_{min} – extended energy usability using the minimum energy in a battery container, in kWh
c_d – cost of unit energy, in NOK/kWh
c_o – fixed cost per recharging activity; refers to charger startup and operation in NOK
c_e – variable recharging costs; refers to the electricity costs, in NOK/hour;
c_{f1} – fixed costs of a charger; includes purchase and installation costs, in NOK

c_{f2} – fixed costs of a charging station; includes cost of land and construction, in NOK
c_{m1} – maintenance cost per charger, in NOK
c_{m2} – maintenance cost per charging station, in NOK
K_n – the number of candidate chargers in a charging station, $n \in N$
$\bar{D}$ – the number of operating days per year
$\bar{M}$ – a sufficiently large positive number
α – annualized factor

Decision Variables

X_i^n – 1 if the mobile battery container from trip $i \in S$ is recharged at charging station $n \in N$, 0 otherwise

Y_{it}^{nk} – 1 if the mobile battery container from trip $i \in S$ starts being recharged on the k^{th} charger at charging station $n \in N$ at time $t \in T$; 0 otherwise, $k = 1, \dots, K_n$

E_i – extended energy using remaining onboard energy at the end of trip $i \in S$, in kWh

Z_{nk} – 1 if the k_{th} charger at charging station n ∈ N is used; 0 otherwise, $k = 1, \dots, K_n$

Z'_k – 1 if charging station $n \in N$ is used; 0 otherwise

Variables Definitions

S – set of scheduled energy use

N – set of candidates charging stations

T – the set of time nodes from the start time of the initial amount of energy to the end time of the final amount of energy

u_n – the charging duration for each recharging activity at the charging station n

$t, t + u_n$– start time and end time for recharging

r_{in} – deadheading travel time from the last site to the charging station

O – set of origin depots

D – set of destination depots

i – origin depot trip, $i \in S \cup O$

j – destination depot trip, $j \in S \cup D$

a_j – start time of the trip i

b_i – end time of the trip j

P – set of trip pairs, such that trip j is served immediately after trip i by the same battery

u_n – the charging duration for each recharging activity at charging station n

A_n – the set of possible recharging activities at charging station n, $(i, j, t) \in A_n$, if $t \geq b_i + r_{in}$ & $t + u_n + r_{in} \leq a_j$

w_{ijnt} – recharging waiting time at the charging station, $w_{ijnt} = t - b_i - r_{in}$

Objective and Constraints Definition

Objective (1) is to minimize the annual total charging system operating costs, which are made up of: deadheading travel costs, recharging costs, recharging waiting costs, charger costs, and charging station costs.

Constraint (2) means that the mobile battery containers cannot be recharged at more than one charging station at the same time.

Constraint (3) means that the extended energy using the remaining energy plus the recharged energy cannot exceed the maximum energy of the mobile battery containers.

Constraint (4) refers to energy conservation.

Constraint (5) ensures that the remaining energy in an electric bus is no less than the minimum energy (i.e., 20% generally).

Constraint (6) gives the relationship of the variables Y to variable X. If a mobile battery container from trip i is recharged at charging station n (i.e., $X_i^n = 1$), there must be a t and a k, which enable $Y_{it}^{nk} = 1$; otherwise, all $Y_{it}^{nk} = 0$.

Constraint (7) assures that each individual charger can only recharge one mobile battery container at a time; in other words, the charging station capacity limitation must be met.

Constraint (8) gives the relationship of the variables Y to variable Z. There must be no recharging activity on a charger if it is not being used; if there is at least one recharging activity on a charger, it must be used. To ensure that there are enough recharging activities, a suitably large number $\overline{M}$ should be utilized as the cardinality for recharging activity set A_n (e.g., a total number of recharging activities) to represent the logical link.

Constraint (9) states the logical relationship between variables Z and Z'; If a charging station is not used, all chargers at that charging station must be turned off; if at least one charger in a charging station is used, the charging station must be turned on.

Constraint (10) imposes an order constraint, stating that the k^{th} charger at charging station n cannot be used until the $(k-1)^{th}$ charger is used.

Constraints (11) – (14) are binary constraints.

Constraint (15) sets an initial range for a mobile battery container at the depot.

4 Methodologies

4.1 Mixed Integer Linear Programming

4.1.1 Introduction

Mixed-integer linear programming (MILP) theory and practice have advanced greatly over the past 50 years, and it is now a vital tool in business and engineering. MIP's success can be attributed to two factors: linear programming (LP) based solvers and MILP's modeling flexibility. MILP has been used to model a wide range of applications since its early stages, and we now have numerous incredibly effective state-of-the-art solvers that incorporate several advanced techniques. [19]

4.1.2 General Formulation

The general form of an integer linear program in the canonical form is:

$$\begin{gathered} maximize\ c^T x \\ subject\ to\ Ax \leq b, \\ x \geq 0, \\ and\ x\ \in Z^n \end{gathered}$$

The general form of an integer linear program in standard form is:

$$\begin{gathered} maximize\ c^T x \\ subject\ to\ Ax + s \leq b, \\ s \geq 0, \\ x \geq 0, \\ and\ x\ \in Z^n \end{gathered}$$

Where, x is a vector that is to be decided and A is a matrix where all entries as integers and c, b as vectors. [20]

4.1.3 Algorithm and Solution Approach

4.1.3.1 Branch and Bound

Branch and bound algorithms are a vast class of algorithms that underpin almost all modern software for solving MILPs. Branch and bound is a divide and conquer strategy that divides the original problem into a number of smaller subproblems and then solves each subproblem recursively. There are four important parts of a branch-and-bound algorithm. A method for obtaining a lower limit on the objective function value of an optimal solution to a given subproblem is known as the lower bounding method. A method for establishing an upper bound on the ideal solution value is known as the upper bounding method. A process for splitting a subproblem into two or more offspring is known as the branching method. A process for choosing the search order is known as a search strategy. [21]

4.1.3.2 Branch and Cut

Integer linear programs are linear programming problems in which some or all of the unknowns are constrained to integer values. Branch and cut is a combinatorial optimization method for solving integer linear programs. Running a branch and bound method and employing cutting planes to tighten the linear programming relaxations is known as a branch and cut [22]. The method uses the conventional simplex algorithm to solve the linear problem without the integer constraint. When an optimal solution is found, and the variable that should be integer has a non-integer value, a cutting plane approach can be used to find additional linear constraints that are satisfied by all possible integer points but violated by the existing fractional solution. These inequalities can be introduced to the linear program, resulting in a different solution that is presumably less fractional when it is resolved [23].

Various types of branching heuristics can be used in branch and cut algorithms but below mentioned branching strategies involve branching on a variable [24].

- Most infeasible branching
- Pseudo cost branching
- Strong branching

4.1.3.3 Branch and Price

Branch and price is an approach in which each node of the search tree can have columns added to the linear programming relaxation (LP relaxation). To reduce the computational and memory requirements, sets of columns are removed from the LP relaxation at the start of the method, and then columns are added back in when needed. The technique is based on the observation that in any optimal solution for large problems, the majority of columns will be non-basic and have their associated variable equal to zero. As a result, a large proportion of the columns are rendered ineffective in resolving the issue [25].

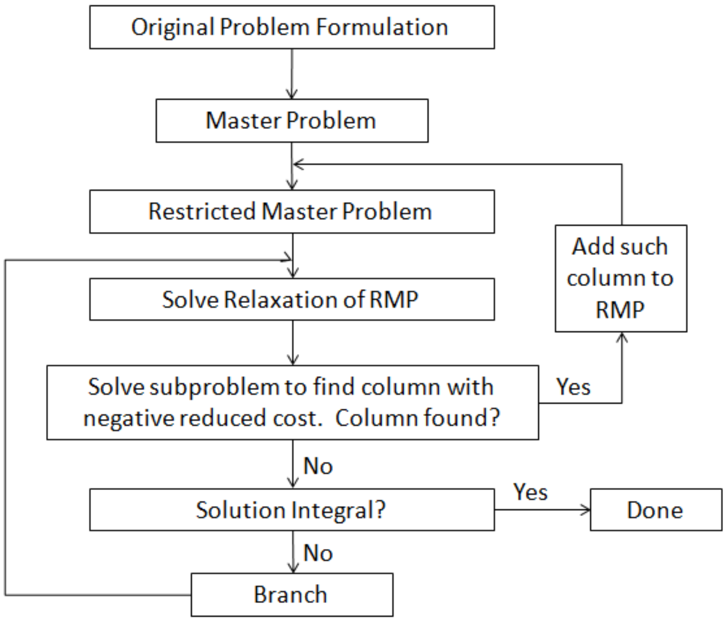

Figure 4-1: Branch and Price Algorithm [26]

4.1.4 Solver Frameworks

A solver framework is a branch-and-bound, branch-and-cut, or branch-and-price algorithm that allow the user to customize some features of the method. For example, the user could want to include a custom branching rule or valid inequalities that are relevant to the problem. To tackle MILP problems, there are a variety of non-commercial software and solvers available. Table 4-1 shows the algorithmic aspects of eight popular solvers, including whether they have a preprocessor, can dynamically produce correct inequalities, can do column formation, have primal heuristics, and what branching and search techniques they have. [21]

Table 4-1: Algorithmic Features of Solvers

	Pre-proc	Built-in Cut Gen	Column Gen	Primal Heuristic	Branching Rules	Search Strategy
ABACYS	No	No	Yes	No	f, h, s	$b, r, d, 2(d, b)$
BCO	No	No	Yes	No	f, h, s	$h(d, b)$
bonsai	No	No	No	No	p	$h(d, b)$
CBC	Yes	Yes	No	Yes	e, f, g, h, s, x	$2(d, b)$
GLPK	No	No	No	No	i, p	b, d, p
lp_solve	No	No	No	No	e, f, i, x	$d, r, e, 2(d, r)$
MINTO	Yes	Yes	Yes	Yes	e, f, g, p, s	$b, d, e, h(d, e)$
SYMPHONY	No	Yes	Yes	No	e, f, g, p, s	$b, r, d, h(d, b)$

Where,

e – pseudo cost branching

f – branching on the variables with the largest fractional part

g – GUB branching

h – branching on hyperplanes

i – branching on the first or last fractional variable (by index)

p – penalty method

s – strong branching

x – SOS (2) branching and branching on semicontinuous variables

b – best-first

d – depth-first

e – best-estimate

p – best-projection

r – breadth-first

$h(b,d)$ – a hybrid method switching from strategy b to d

$h(b,e)$ – a hybrid method switching from strategy b to e

$2(d,b)$ – a two-phase method switching from strategy d to b

$2(d,r)$ – a two-phase method switching from strategy d to r

4.1.5 Applications

When modeling problems as a linear program, there are two key reasons to use integer variables. The first is an integer variable, which represents only integer amounts. Building 3.7 vehicles, for example, is not possible. The second type of variable is an integer variable, which represents decisions (such as whether to include an edge in a graph) and should only have the values 0 or 1. As a result of these considerations, integer linear programming can be utilized in a variety of applications, some of which are briefly detailed below. [27]

1. The applications
2. Production Planning
3. Scheduling
4. Territorial Partitioning
5. Telecommunication Networks
6. Cellular Networks
7. Cash-Flow Matching
8. Energy System Optimization
9. UAV Guidance

4.2 Mixed Integer Non-Linear Programming

4.2.1 Introduction

Mixed-Integer Nonlinear Programming (MINLP) is a type of mathematical programming that uses continuous and discrete variables as well as nonlinearities in the objective function and restrictions. Many optimal choice issues in science, engineering, and public sector applications involve both discrete decisions and non-linear system dynamics that influence the ultimate design or plan. The combinatorial difficulty of optimizing over discrete variable sets is combined with the complexities of dealing with nonlinear functions in mixed-integer nonlinear programming (MINLP) situations. [28]

4.2.2 General Formulation

The general form of an MINLP is:

$$\begin{gathered} minimize\ f(x,y) \\ subject\ to\ g(x,y) \leq 0 \\ x \in X \\ y \in Y\ integer \end{gathered}$$

The functions $f(x, y)$ and $g(x, y)$ are nonlinear objective functions and nonlinear constraint functions, respectively. The decision variables are x and y, with y having to be integer-valued. X and Y are variables with bounding-box constraints. [29]

4.2.3 Algorithm

MINLP issues are extremely difficult to solve precisely because they contain all of the challenges of their subclasses: the combinatorial character of mixed-integer programming (MIP) and the difficulty in solving nonconvex (and even convex) nonlinear programs (NLP). Because the subclasses MIP and NLP belong to the class of theoretically difficult problems (NP-complete), it is not surprising that solving MINLP can be a tough and risky endeavor. Fortunately, the MIP and NLP component structure within MINLP provides a collection of natural algorithmic approaches that utilize the structure of each of the subcomponents. [28]

4.2.4 Solution Approach

Innovative methodologies and related techniques borrowed and enhanced from MIP are used to solve MINLPs. Since the early 1980s, the literature has studied Outer Approximation (OA) Methods, Branch-and-Bound (B&B), Extended Cutting Plane Methods, and Generalized Bender's Decomposition (GBD) for solving MINLPs. In general, these techniques rely on the sequential resolution of closely related NLP problems. For example, B&B begins by forming a pure continuous NLP problem by removing the discrete variables' integrality requirements (often called the relaxed MINLP or RMINLP). Furthermore, each node of the growing B&B tree represents an RMINLP solution with updated bounds on the discrete variables. [29]

4.2.5 Applications

In optimization, MINLP can be considered one of the most general modeling paradigms with subproblems encompassing both nonlinear programming and mixed-integer linear programming. MINLPs have been employed in a wide range of applications, including the process industry, finance, engineering, management science, and operations research. It involves issues with process flow sheets, portfolio selection, batch processing in chemical engineering (including mixing, reaction, and centrifuge separation), and the best design of gas or water transmission networks. Automobile, aviation, and VLSI production are among the other fields of interest. [28]

4.3 Genetic Algorithm

4.3.1 Introduction

A genetic algorithm (GA) is a natural selection-inspired metaheuristic that is part of the larger class of evolutionary algorithms in computer science and operations research. GA, which rely on biologically inspired operators such as mutation, crossover, and selection, are often employed to develop high-quality solutions to optimization and search problems. It becomes the most popular and successful optimization algorithm in the theory of artificial intelligence optimization methods due to its great optimization-searching performance. The essential notion behind the GA is that an initial set of chromosomes (solutions) is formed as a population and exposed to fitness function evaluation (scaled from the objective function). Chromosomes with a high level of fitness are more likely to survive and reproduce. They are chosen as parent chromosomes to reproduce offspring chromosomes, whereas those with low fitness ratings are eliminated. Following the crossover and mutation of parent chromosome pairs, offspring chromosomes are generated, resulting in a new generation with not just enhanced genes, but also some of their parents' features. In a novel reproductive mechanism, the chromosomes with relatively high fitness levels survive and reproduce. The reproductive and evolutionary process is repeated over numerous generations, with chromosomes changing to produce high-quality offspring and population size reducing until a chromosome with the best fitness is discovered. This chromosome with the highest fitness is the ideal solution being sought. [30] [31]

4.3.2 Processing Step and Algorithm

The genetic algorithm is a sophisticated intelligent optimization method. It is based on Darwin's theory of evolution, which claims that evolution selects the superior while removing the worse and that the good fitness survives. The overall premise of GA is simple to grasp, but the detailed methods and processes used by GA to solve optimization problems are extremely complex. Instead of studying the GA itself, the major goal of this thesis is to study vehicle schedule optimization problems using Genetic Algorithms, therefore powerful software like MATLAB or Excel is introduced to conduct the calculation of searching for an optimal solution for the GA-based method. [32]

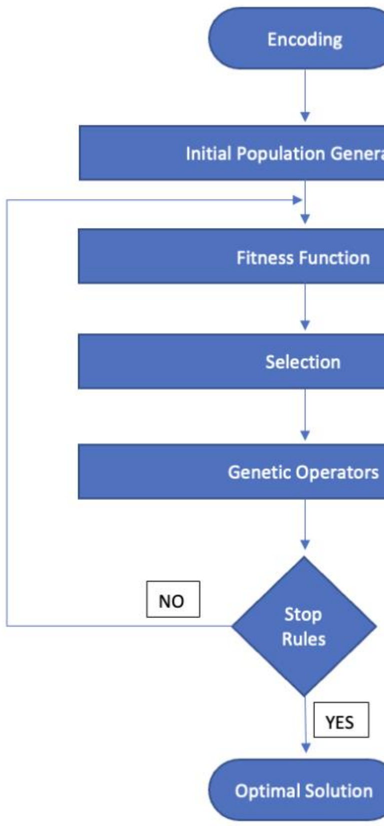

Figure 4-2: Genetic Algorithm

The following points defines the processing steps used in GA [33][31].

1. Population and Size - The starting population is produced by selecting plausible issue solutions at random.
2. Encoding Scheme - The chromosome is made up of genes that must be encoded in such a way that they can represent a solution to the intended problem.

3. Fitness Function - The fitness function is derived from the objective function and scaled. It is intended to assess an individual's fitness worth.
4. Selection Policy - The selection policy selects chromosomes with higher fitness to reproduce to ensure that the great genes are passed down to future generations. It focuses on two techniques of selection: roulette wheel selection and tournament selection.
5. Genetic Operators - Crossover and mutation are examples of genetic operators. They mimic the reproductive process of the chosen parent chromosomes growing into offspring chromosomes.
6. Stop Rules/Criterion - When the reproductive process is complete, the Stop rules are determined. As a stop condition, the maximum number of generations is frequently set.

4.3.3 Pros and Cons

This method excels at finding global optimizations. In real life, it can be utilized to efficiently handle some highly complex optimization problems. However, despite its widespread use, there are certain limitations, such as selection of initial population, a lack of strength in looking for local optimizations, premature convergence, selection of efficient fitness functions, degree of mutation and cross over, selection of encoding schemes and its solution is influenced by experience-based parameter settings. [34]

4.3.4 Applications

Combinatorial optimization primarily seeks to maximize productivity while employing limited resources, while also satisfying a variety of additional restrictions, such as the bin-packing problem, vehicle routing problem, and airline crew scheduling problem. The applications include facility layout, scheduling, inventory control, forecasting and network design, information security, image and video processing, medical imaging, precision agriculture, gaming, wireless networking, load balancing, localization, bandwidth, and channel allocation. The basic goal of multi-objective optimization is to discover the optimal solution for several conflicting objectives under certain limitations, such as the multiple-objective transportation problem and the capacitated plant location problem. [34]

4.4 Large Neighborhood Search Algorithm

4.4.1 Introduction

Neighborhood search is a mathematical optimization strategy that aims to find good or near-optimal solutions to a combinatorial optimization problem by repeatedly changing a current solution into a different solution in its neighborhood. A solution's neighborhood is a group of comparable solutions that can be acquired by making minor changes to the original solution. The Large Neighborhood Search (LNS) proposed by Shaw [35] can be defined as a metaheuristic that finds an initial solution that is refined over time by repeatedly deleting and reconstructing it. The primary idea behind LNS is to conduct searches in large neighborhoods, which may contain more and maybe better solutions than smaller ones. A destruct method and a repair method define the neighborhood of a solution implicitly. A destroy technique destroys a portion of the present solution, whereas a repair method rebuilds it. [15]

4.4.2 Algorithm

The pseudo-code for the LNS heuristics is [36]:

1: input: a feasible solution x
2: $x^b = x$;
3: **repeat**
4: $\quad x^t = r(d(x))$;
5: $\quad$ **if** accept (x^t, x) **then**
6: $\qquad x = x^t$;
7: $\quad$ **end if**
8: $\quad$ **if** $c(x^t) < c(x^b)$ **then**
9: $\qquad x^b = x^t$;
10: $\quad$ **end if**
11: **until** the stop criterion is met
12: **return** x^b

Where,

x – current solution

x^b – the best solution

x^t – a temporary solution

$d(x)$ – a destroy function that destroys a copy of x

$r(d(x))$ – a repair function that returns a feasible solution built from the destroyed one

$c(x^t)$ – the objective value of solution x^t

$c(x^b)$ – the objective value of solution x^b

The global best solution is initialized in line 2. To find a new solution, the heuristic uses the destroy method first, then the repair method in line 4. The new answer is examined at line 5, and in line 6 the heuristic determines whether it should be accepted as the new current solution or rejected. The accept function can be used in a variety of ways. The simplest option is to accept only improving solutions. Line 8 determines whether the new answer is superior to the best-known alternative. If necessary, the optimal answer is updated in line 9. The termination condition is tested in line 11. The best solutions found are returned in line 12, after the termination criterion is met or the iteration limit is reached.

4.4.3 Adaptive Large Neighborhood Search

Adaptive large neighborhood search (ALNS) is an extension of LNS which uses multiple neighborhoods within the same search. It was first proposed by S. Ropke and D. Pisinger [37]. It improves on the LNS heuristic by allowing the employment of several destroy and repair methods in the same search. Each destroy/repair technique has a weight assigned to it, which determines how often it is used during the search. The weights are dynamically updated as the search advances, allowing the heuristic to adapt to the situation at hand and the current stage of the search. The pseudo-code for the LNS heuristics is [37]:

input: a feasible solution x
$x^b = x; \rho^- = (1, \ldots, 1); \rho^+ = (1, \ldots, 1);$
repeat
    select destroy and repair methods $d \in \Omega^-$ and $r \in \Omega^+$ and $\rho^- \in \rho^+$;
    $x^t = r\big(d(x)\big);$
    if accept (x^t, x) **then**
        $x = x^t;$
    end if
    if $c(x^t) < c(x^b)$ **then**
        $x^b = x^t;$
    end if
    update ρ^- and ρ^+;
until stop criterion is met
return x^b

Where,

$\Omega^-$and Ω^+ – sets of destroy and repair methods respectively

ρ^- and ρ^+ – variables to store the weight of each destroy and repair method respectively

4.5 MATLAB Optimization Toolbox

MATLAB Optimization Toolbox is a MathWork's software package for optimization. It is a MATLAB add-on product that was first made available for MATLAB in 1990. It provides a library of solvers that can be utilized from within the MATLAB environment that includes functions for determining parameters that minimize or maximize objectives while meeting restrictions. [38]

Optimization Toolbox has algorithms for:
1. Linear Programming
2. Mixed-Integer Linear Programming
3. Quadratic Programming
4. Nonlinear Programming
5. Linear Least Squares
6. Nonlinear Least Squares
7. Nonlinear Equation Solving
8. Multi-Objective Optimization

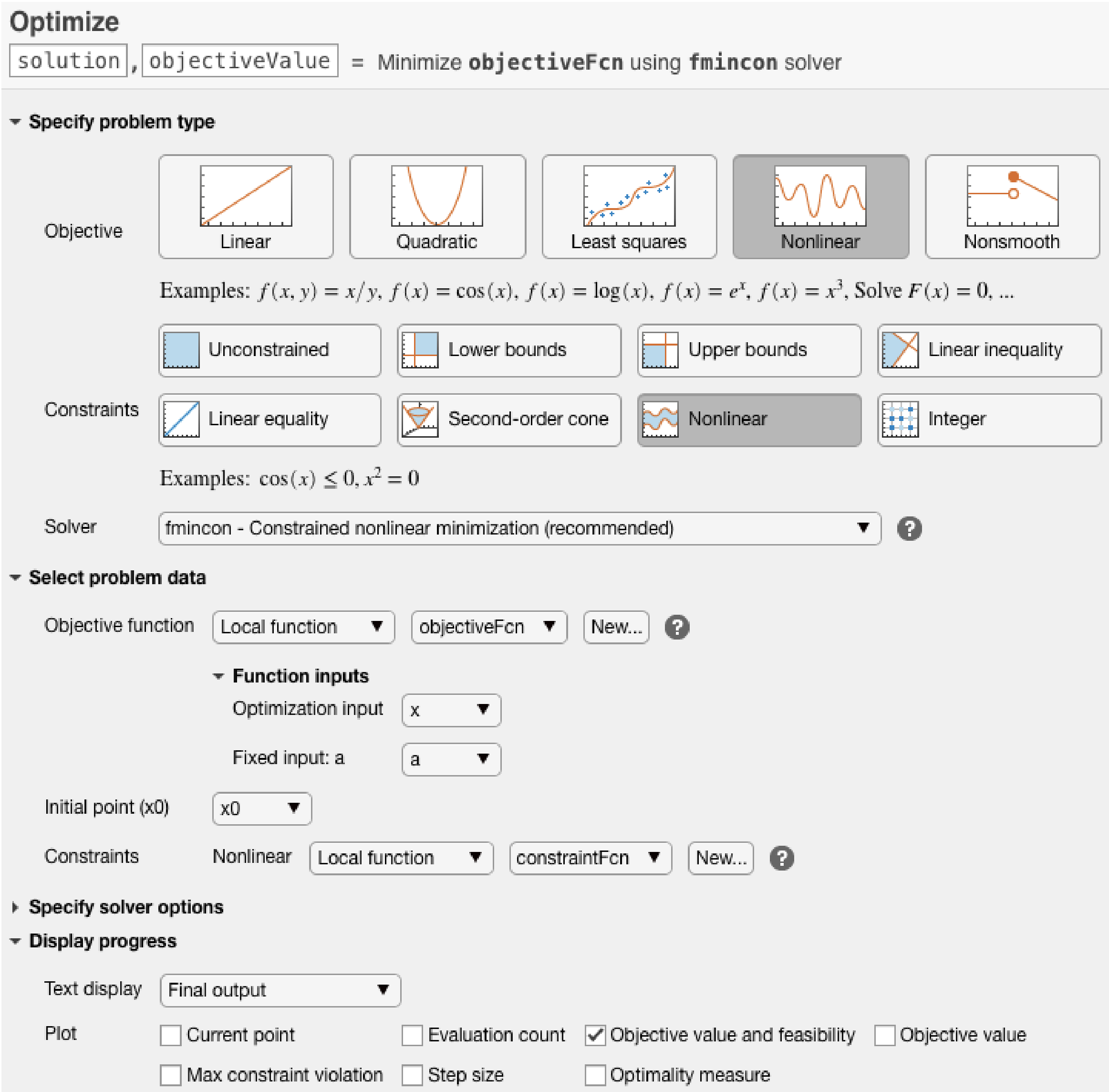

Figure 4-3: MATLAB Optimization Toolbox User Interface

The variable expressions that reflect the underlying mathematics or define your optimization problem with functions and matrices can be specified. For faster and more accurate answers, employ automatic differentiation of objective and constraint functions can be employed. The toolbox solvers can be used to identify optimal solutions to continuous and discrete problems, conduct tradeoff evaluations, and incorporate optimization approaches into algorithms and applications. You may use the toolbox to conduct design optimization tasks like parameter estimates, component selection, and parameter tuning. It allows you to find the best solutions in applications like portfolio optimization, energy management and trading, and production planning. [38]

Some of the applications of this toolbox are:

1. Engineering Optimization
    a. Optimal Control
    b. Optimal Mechanical Design
2. Parameter Estimation
    a. Material Parameter Estimation
    b. Estimation of coefficients of ODE's

3. Computational Finance
    a. Portfolio Optimization
    b. Cashflow Matching
4. Utilities and Energy
    a. Security Constraints for Optimal Power Flow
    b. Power Systems Analysis

4.6 Microsoft Excel Solver

Microsoft Excel Solver is an optimization tool that can be used to identify how to change the assumptions in a model to reach the desired output. It is a Microsoft Excel add-in tool that allows you to perform what-if analysis. It can be used to determine an optimal (highest or minimum) value for a formula in a single cell, known as the objective cell, that is constrained or limited by the values of other formula cells on a worksheet. [39]

It operates on a set of cells known as decision variables or simply variable cells, which are used to compute the formulas in the objective and constraint cells. The values in the choice variable cells are adjusted by the solver to satisfy the constraints on the constraint cells and give the desired outcome for the objective cell. Simply put, the solver can be used to determine the maximum or lowest value of one cell by changing the values of other cells. For example, you can alter the size of your expected advertising expenditure to observe how it affects your projected profit.

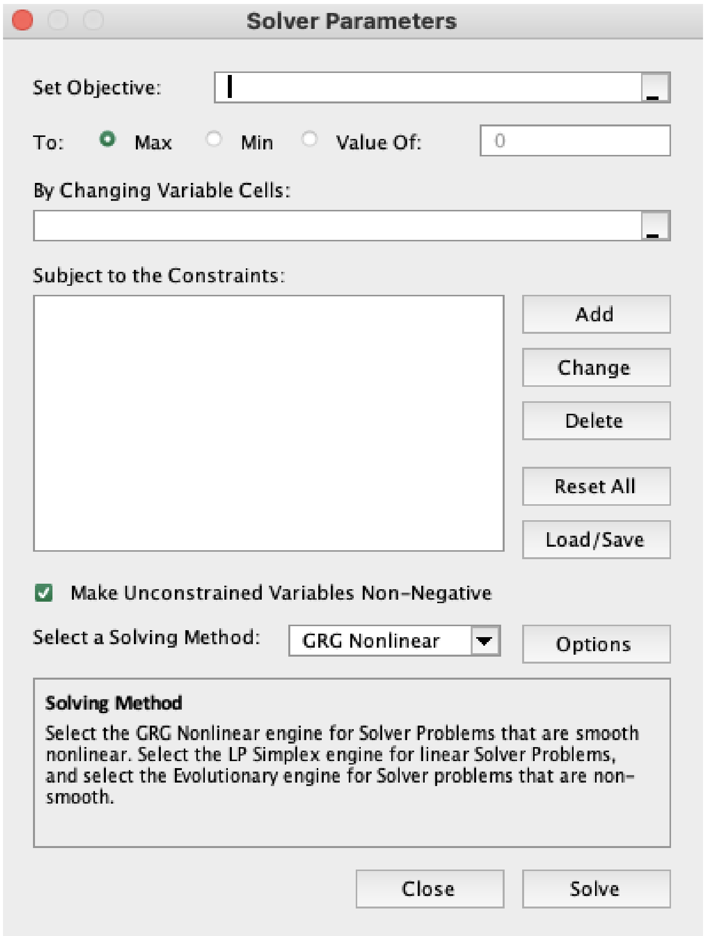

Figure 4-4: Excel Solver User Interface

Since the free version of the Excel solver is limited to 200 decision variables so OpenSolver is used in this project. OpenSolver is an open-source optimization tool for Microsoft Excel with very close resemblance to all the functionalities that excel solver provides but without the restriction on the number of variables. The algorithm used in this project is COIN-OR CBC (Linear Solver). The COIN Branch and Cut solver (CBC) is the default solver for OpenSolver and is an open-source mixed-integer program (MIP) solver.

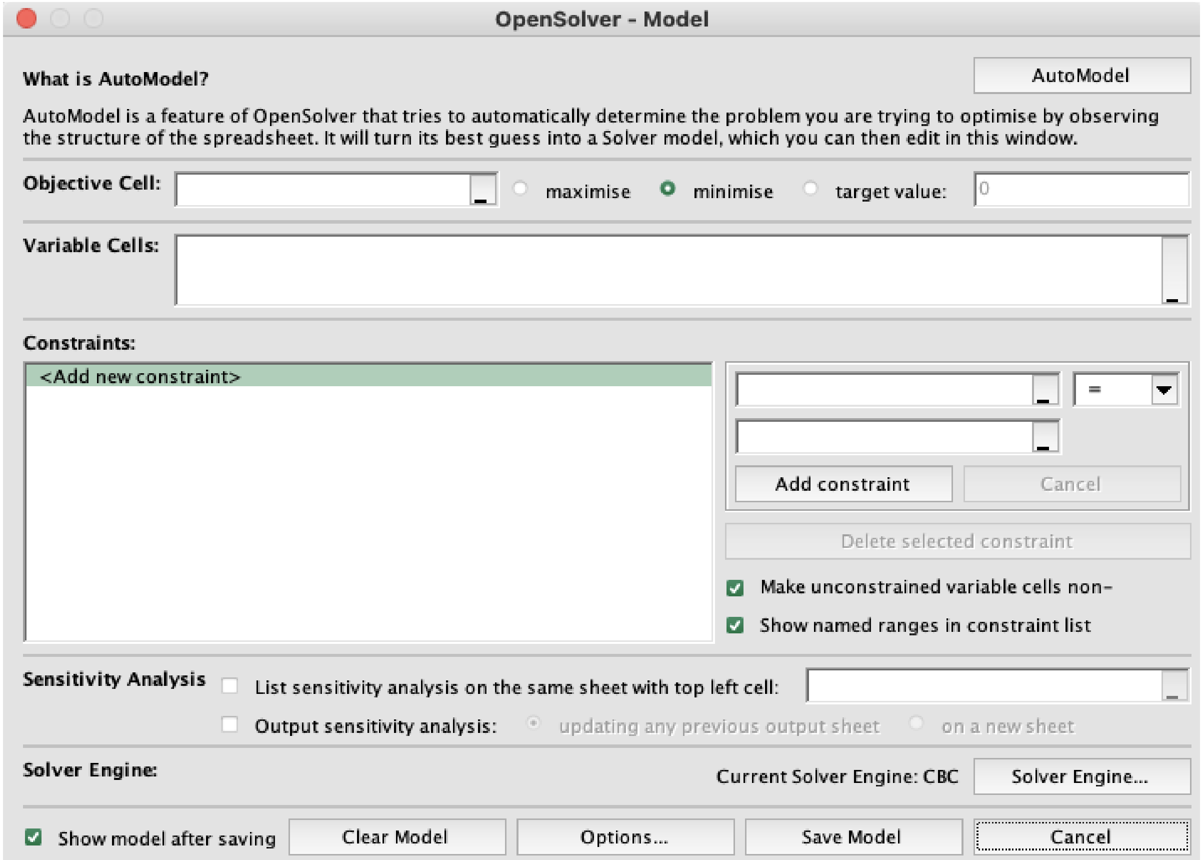

Figure 4-5: OpenSolver User Interface

5 Case Studies

5.1 VRP – Energy Supply and Demand

5.1.1 Problem Description

For this case scenario simulation, a scenario in which a mobile battery container distributor company intends to place N number battery containers in the deport and supply to the construction site according to the demand. There are normally four types of equipment for this case, small equipment, auxiliary equipment, excavator, and mobile crane with their own energy requirements of the rated machine.

The following decisions are to be made using the optimization function.

- Total distance traveled
- Total energy demand
- Total number of mobile battery containers

5.1.2 Problem Definition

A distributor needs to send a vehicle with battery containers to construction sites. The following criteria should be met for optimal routing of the vehicle.

- The demand of each machine in one must be satisfied by one battery container.
- Container capacity must not be exceeded for any vehicle.
- The total distance should be minimized.

For this, all the position of the site is considered in a 100×100 grid where the initial point(depot) is in the middle of the grid i.e., (50,50) which is considered as point 1, and the rest of the points n with their location (X, Y), demand D_i and description are tabulated in Table 5-1.

There are n = 30 points where the machines are located and the distance between node i and node j is d_{ij} which is scaled in the unit as 1 unit in the grid. There is an N number of the homogenous vehicle each with a limited capacity C = 576 kWh which fulfills the demand of each point. Exactly one vehicle visits each point that satisfies the demand of that point without exceeding the capacity of the vehicle. There are two decision variables x_{ij} which are binary and are assigned '1' if a truck goes from node i to node j and f_{ij} is an integer which is the number of remaining energy units in a truck going from node i to node j

The number of vehicles required to fulfill the demand, the minimum possible distance traveled, and the flow of the vehicle from node i to node j is calculated using MILP in Excel Solver.

Table 5-1: Demand of Each Vehicle

Points	Location (X, Y)	Demand (kWh)	Description
1	(50, 50)	-	Depot
2	(92, 2)	50	Small Equipment
3	(4, 86)	100	Auxiliary Equipment
4	(71, 30)	100	Auxiliary Equipment
5	(36, 35)	250	Excavator
6	(20, 93)	500	Mobile Crane
7	(65, 95)	50	Small Equipment
8	(80, 75)	50	Small Equipment
9	(44, 8)	50	Small Equipment
10	(55, 47)	250	Excavator
11	(91, 52)	250	Excavator
12	(90, 87)	100	Auxiliary Equipment
13	(31, 70)	100	Auxiliary Equipment
14	(90, 73)	500	Mobile Crane
15	(82, 32)	500	Mobile Crane
16	(17, 83)	500	Mobile Crane
17	(87, 58)	500	Mobile Crane
18	(84, 9)	250	Excavator
19	(97, 46)	250	Excavator
20	(60, 17)	250	Excavator
21	(72, 45)	250	Excavator
22	(92, 29)	50	Small Equipment
23	(73, 64)	50	Small Equipment
24	(21, 10)	50	Small Equipment
25	(12, 35)	100	Auxiliary Equipment
26	(7, 43)	100	Auxiliary Equipment
27	(43, 52)	100	Auxiliary Equipment
28	(86, 87)	250	Excavator
29	(27, 92)	250	Excavator
30	(29, 56)	250	Excavator
31	(7, 24)	250	Excavator

5.1.3 Mathematical Formulation

The mathematical formulation used in this study case is introduced in [40].

Objective Function:

$$\min \sum_{i=1}^{n} \sum_{j=1}^{n} d_{ij} x_{ij} \tag{16}$$

Constraints:

$$\sum_{j=1}^{n} x_{ij} = 1 \qquad \forall i = 2, \dots, n \tag{17}$$

$$\sum_{j=1}^{n} x_{ji} = 1 \quad \forall i = 2, \dots, n \tag{18}$$

$$\sum_{j=1}^{n} f_{ji} - \sum_{j=1}^{n} f_{ij} = D_i \quad \forall i = 2, \dots, n \tag{19}$$

$$0 \le f_{ij} \le C x_{ij} \quad \forall i, j = 1, \dots, n \tag{20}$$

$$x_{ij} \in \{0,1\} \quad \forall i, j = 1, \dots, n \tag{21}$$

Parameters

n – number of points (1 - depot, $2, \dots, n$ - clients)

d_{ij} – distance from node i to node j

D_i – demand of client i

C – capacity of each truck

Variables

x_{ij} – 1 if a truck goes from node i to node j (binary)

f_{ij} – number of units in a truck going from node i to node j

Where,

Equation (16) represents the objective function which is the sum product of the binary variable and distance traveled between node i and node j that minimizes the total distance traveled to fulfill all the demands.

Equation (17) and (18) represents the binary variable constraints that must be 1 for the route from node i to j and j to i respectively.

Equation (19) states that the number of energy units remaining in the vehicle after a trip should be equal to the demand of the next point so that it can satisfy the next demand point.

Equation (20) states that any demand point should not exceed the total capacity of the vehicle.

Equation (21) simply means the decision variable x_{ij} is a binary variable.

5.1.4 Results and Interpretation

The number of vehicles required to fulfill all the demands is N = 14 and the route of the vehicle from the depot to respective points is shown in Figure 5-1. All the points are fulfilled by exactly one vehicle with a minimum number of vehicles required to fulfill all the energy demands possible. If the vehicle goes to the point with lower demand, then it fulfills the demand before going to the second shortest demand point and finally returning to the depot for e.g., point 1,7,28,12,28,8,1 which is 50,100,250,50 = 450 kWh of total demand in a single route. If the vehicle goes to the point with higher demand, then it fulfills the demand and comes back to the depot for e.g., point 6 which is 500kWh. And if the fulfills the higher demand like 14 which is

500 kWh and finds an unfilled demand like point 23 then it serves point 23 which is 50 kWh making a total of 550 kWh before coming to the depot.

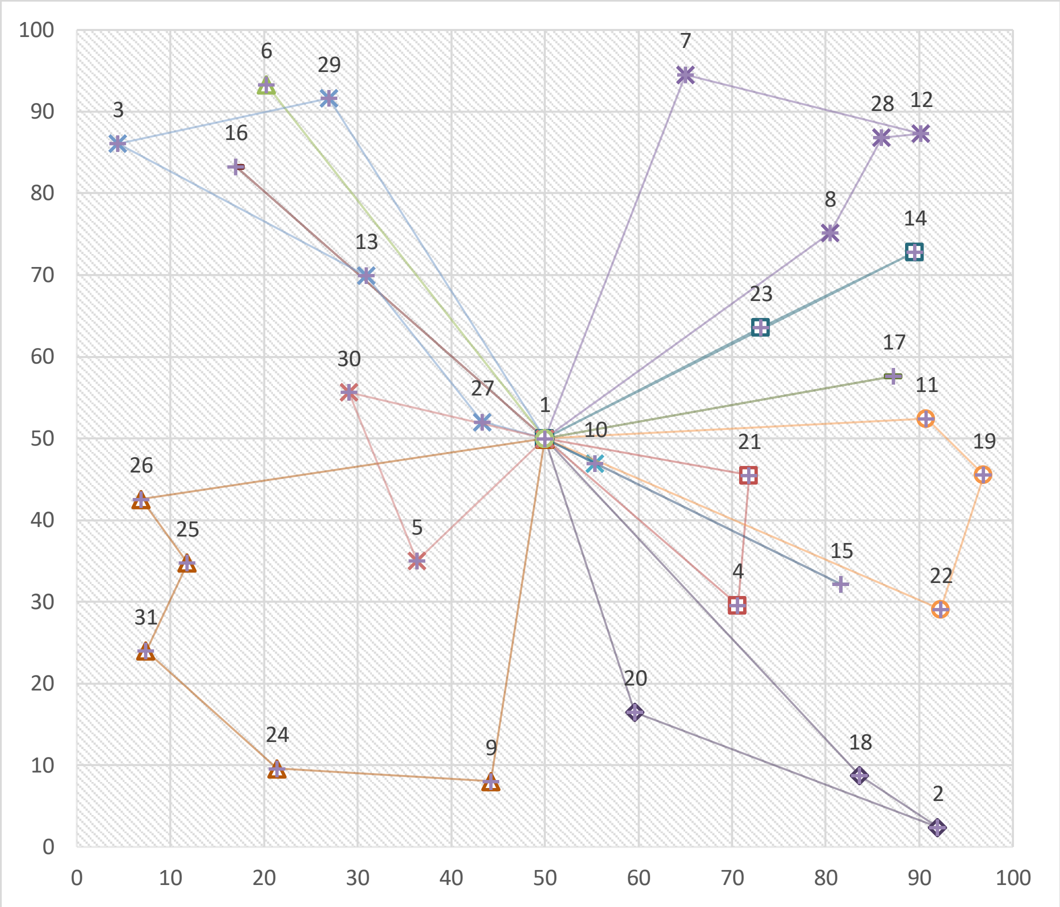

Figure 5-1: VRP Solution (2 mins)

If all the 30 points are electric machinery operating in 4 different construction sites such that all 30 points are divided as i.e., upper left, upper right, lower left, and the lower right plane of the graph, then it can be stated that the optimal solution is feasible for each construction site with their own energy demand.

These kinds of computation problems usually take a long computational time to reach the optimal solution so, the results are compared in Table 5-2 to find the optimal computational time using 8 core processor (Apple M1 chip) in the excel solver. There is very little variation in the outcomes after 2 minutes and to get a noticeable difference the computational time must be greater than 230 minutes which changes the objective value by 1.2 units.

Table 5-2: Effect of Limited Computational Time on Objective Value

Computational Time (min)	Objective Value (distance)
0.5	1258.3
1	1245.5
2	1240.8
5	1240.8
10	1240.8
20	1240.8
230	1239.6
600	1238.2

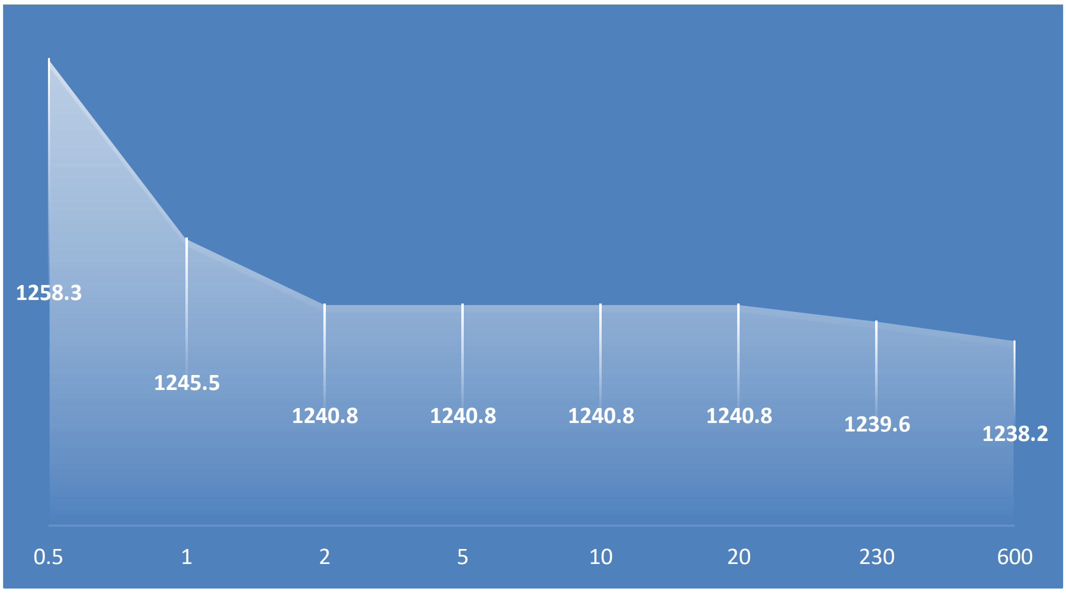

Figure 5-2: Computational Time vs Objective Value

The 230 minutes of computational time causes the route to change from 1-7-12-28-8-1 to 1-7-28-12-8-1 which is 1.2 units shorter than the previous results as shown in Figure 5-2. The solver computed the MILP with the objective function, constraints, and decision variable to find the minimum distance required to travel to fulfill all the demands to be 1240.8 units.

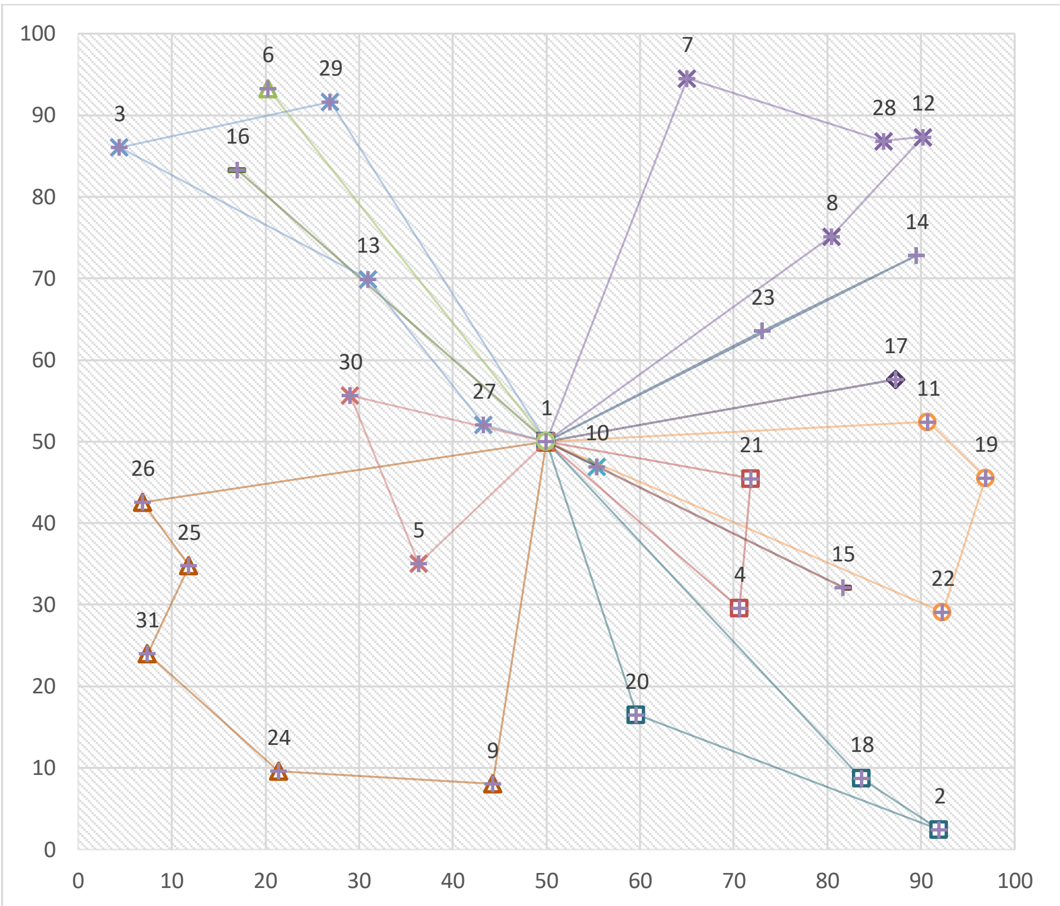
Figure 5-3: VRP Solution (230 mins)

5.2 VSP – Multiple-Depot Scheduling

The vehicle scheduling problem is more complex in practice than in the sample scenario examined in the previous case study. Other more real-life aspects must be considered, as well as numerous uncertainties. In this study, the real-world problem is simplified and constrained to a normal planning cycle, as was the case in the pilot study. The optimization methods developed in the previous chapter will be employed in this chapter to optimize a real scheduling problem for several construction sites.

5.2.1 Problem Description

For a real case scenario, a typical construction site is considered with the following parameters:

- Battery specification:
    1. Weight – 7.5 ton
    2. Energy - 576 kWh
- Mobile battery containers specification:
    1. Length – 45 ft
    2. Breadth - 8 ft
    3. Height - 8 ft 6 inches
- Equipment specification:
    1. Equipment:
        - Mobile Crane – 500 kWh
        - Excavator – 250 kWh
        - Auxiliary Machines – 100 kWh
        - Diverse Small Machines – 50 kWh
- The combined charging system (CCS) type 2 or Combo 2 cable can be used in the charging station as it can provide power at up to 350 kW. The charging time for one battery from 0% to 100% using CCS type 2 is 2 hours.
- Working time from 07:00 – 15:00
- The three possible charging stations:
    1. Hauen
    2. Tømmerkaia
    3. Floodmyrvegen

5.2.2 Problem Definition

The main tasks in this study are first, to schedule the mobile battery container (called 'vehicle' in this chapter) to be used by machineries (called 'customer' in this chapter) in construction site and recharged in charging station (called 'depot' in this chapter) and second, to determine the total cost associated with it which is based on the shortest distance to from charging station to the construction site considering there are three charging stations (depot) and ten machines operated (customers) in five construction sites as shown in Figure 5-4. The latitude and longitude of the depots and customers along with their operating time, delivery amounts are listed in Table 5-3.

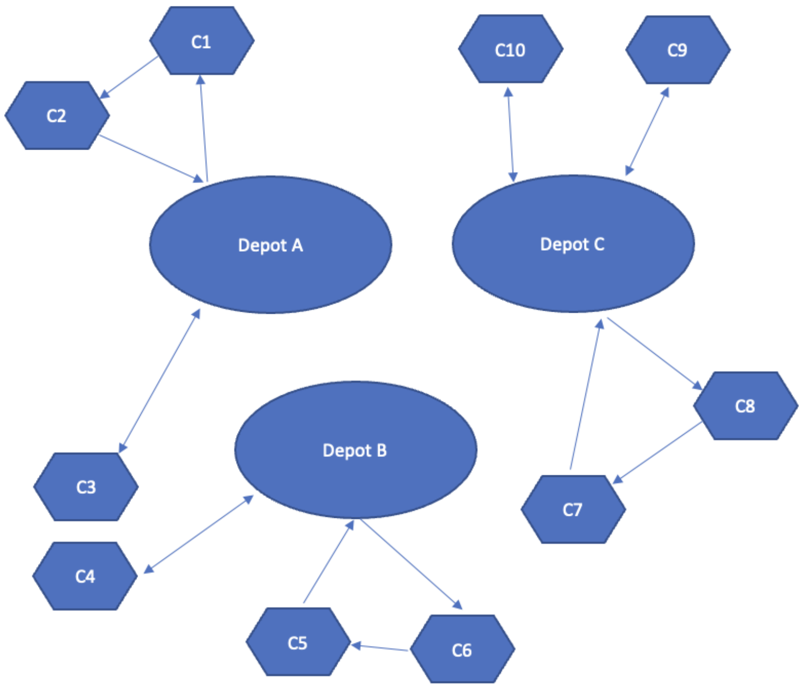

Figure 5-4: Multiple-Depot Scheduling

In mathematical formulations, the following assumptions are made to simplify problems:

1. The vehicle operates according to the previously established timetable.
2. All the vehicles are homogeneous and have the same energy storage.
3. The charge consumed is proportional to the working hours by the customers on the construction site.
4. The number of vehicles in each depot is manually set, and then the solver determines the number of vehicles to the route.
5. The time it takes to charge is directly proportional to the amount of energy recharged.
6. All chargers are homogeneous fast chargers. The recharging duration is fixed (2 hours), and continuous.

Table 5-3: Address and GPS co-ordinates of depot and customer

Name	Address	Latitude	Longitude
Depot A	Floodmyrvegen	59.12	9.69
Depot B	Hauen	59.17	9.64
Depot C	Tømmerkaia	59.20	9.61
Customer 1	Gulset	59.22	9.56
Customer 2	Gulset	59.22	9.56
Customer 3	Herøya	59.11	9.65
Customer 4	Herøya	59.11	9.65
Customer 5	Vallermyrvegen	59.14	9.67
Customer 6	Vallermyrvegen	59.14	9.67
Customer 7	Skotfoss	59.21	9.53
Customer 8	Skotfoss	59.21	9.53
Customer 9	Hoppestad	59.25	9.57
Customer 10	Hoppestad	59.25	9.57

Table 5-4: Time windows, service time, and delivery amount of all the customers

Name	Time Window Start (hh:mm)	Time Window End (hh:mm)	Service Time (Hours)	Delivery Amount (kWh)
Customer 1	07:00	15:00	7	500
Customer 2	11:00	15:00	3.5	250
Customer 3	07:00	15:00	7	500
Customer 4	07:00	11:00	3.5	250
Customer 5	07:00	11:00	4	250
Customer 6	11:00	15:00	3.5	250
Customer 7	07:00	11:00	2	150
Customer 8	11:00	15:00	3	150
Customer 9	07:00	15:00	7	500
Customer 10	17:00	11:00	3.5	100

The profit collected is assumed to be 5x the delivery amount i.e., 5 NOK for 1 kWh. Bing maps driving distance (km) is used as the distances computational method and Bing maps driving duration is used as the duration computation method between the depot and the customer. As stated earlier in the assumptions, the vehicles are homogenous with a capacity of 576 kWh.

There are three vehicles each in all three depots as shown in Table 5-4, which serves the energy need of all the customers. The fixed cost per trip is assumed to be 1000 NOK and the cost per unit (km) distance is assumed to be 10 NOK.

Table 5-5: Vehicle name with their corresponding depot name

Depot	Vehicle
A	V1 V2 V3
B	V4 V5 V6
C	V7 V8 V9

5.2.3 Mathematical Formulation

The mathematical formulation used in this study case is introduced by Erdoğan [41].

Maximize

$$\sum_{i \in V_C} \sum_{k \in K} p_i y_i^k - \sum_{(i,j) \in A} \sum_{k \in K} c_{ij}^k x_{ij}^k - \sum_{j \in V_C} \sum_{k \in K} f^k x_{o^k,j}^k - \pi \sum_{i \in V} v_i \tag{22}$$

Subject to

$$\sum_{k\in K} y_i^k = 1 \quad \forall i \in V_M, \tag{23}$$

$$\sum_{k\in K} y_i^k \leq 1 \quad \forall i \in V_C \backslash V_M, \tag{24}$$

$$\sum_{j\in V\backslash\{i\}} x_{ij}^k \leq \sum_{j\in V\backslash\{i\}} x_{ji}^k \quad \forall i \in V_C, k \in K, \tag{25}$$

$$\sum_{p\in S, q\in V\backslash S} x_{pq}^k \geq y_i^k \quad \forall i \in V_C, k \in K, S \subset V: o^k \in S, i \in V\backslash S, \tag{26}$$

$$\sum_{p\in S, q\in V\backslash S} x_{pq}^k \geq \Omega y_i^k \quad \forall i \in V_C, k \in K, S \subset V: i \in S, r^k \in V\backslash S, \tag{27}$$

$$\sum_{j\in V_C} x_{o^k,j}^k \leq 1 \quad \forall k \in K, \tag{28}$$

$$\sum_{k\in K} x_{ij}^k \leq 1 - \beta \quad \forall (i,j) \in A: q_i > 0 \ and \ \hat{q}_j > 0, \tag{29}$$

$$\sum_{j\in V\backslash\{i\}} w_{ij}^k - \sum_{j\in V\backslash\{i\}} w_{ji}^k = q_i y_i^k \quad \forall i \in V_C, k \in K, \tag{30}$$

$$\sum_{i\in V_C} w_{i,r^k}^k = \sum_{j\in V_c} q_j y_j^k \quad \forall k \in K, \tag{31}$$

$$\sum_{j\in V\backslash\{i\}} z_{ij}^k - \sum_{j\in V\backslash\{i\}} z_{ji}^k = \hat{q}_i y_i^k \quad \forall i \in V_C, k \in K, \tag{32}$$

$$\sum_{i\in V_C} z_{o^k,i}^k = \sum_{i\in V_c} \hat{q}_i \, y_i^k \quad \forall k \in K, \tag{33}$$

$$t_i^k + \left(\hat{d}_{ij} + s_i\right) x_{ij}^k - W^k\left(1 - x_{ij}^k\right) \leq t_j^k \quad \forall (i,j) \in A: j \in V_C, k \in K, \tag{34}$$

$$a_i \leq t_i^k \leq b_i - s_i + v_i \quad \forall i \in V_C, k \in K, \tag{35}$$

$$v_i \leq M.\Theta \quad \forall i \in V_C, \tag{36}$$

$$t_{o^k}^k = \tau^k \quad \forall k \in K, \tag{37}$$

$$t_i^k + \left(s_i + \hat{d}_{ij}\right) x_{i,r^k}^k \leq b_{r^k} + v_{r^k} + M(1 - \Omega) \quad \forall (i,j) \in A: i \in V_C, k \in K, \tag{38}$$

$$w_{ij}^k + z_{ij}^k \leq Q^k x_{ij}^k \quad \forall (i,j) \in A, k \in K, \tag{39}$$

$$\sum_{(i,j)\in A} d_{ij} x_{ij}^k \leq D^k \quad \forall (i,j) \in A, k \in K, \tag{40}$$

$$\sum_{(i,j)\in A} \hat{d}_{ij} x_{ij}^k \leq \widehat{D}^k \quad \forall (i,j) \in A, k \in K, \tag{41}$$

$$\sum_{i\in V_C} s_i y_i^k + \sum_{(i,j)\in A} \hat{d}_{ij} x_{ij}^k \leq W^k \quad \forall (i,j) \in A, k \in K, \tag{42}$$

$$x_{ij}^k \in \{0,1\} \quad \forall (i,j) \in A, k \in K, \tag{43}$$

$$y_i^k \in \{0,1\} \quad \forall i \in V_C, k \in K, \tag{44}$$

$$v_i \geq 0 \quad \forall i \in V_C, \tag{45}$$

$$w_{ij}^k \geq 0 \quad \forall (i,j) \in A, k \in K, \tag{46}$$

$$z_{ij}^k \geq 0 \quad \forall (i,j) \in A, k \in K, \tag{47}$$

Parameters

V_D	– the vertex to contain the depots
V_C	– the vertex to contain the customers
$V_M \subseteq V_C$	– the set of customers that must be delivered
p_i	– the quantity to pick up per customer
$\hat{q}_i$	– the quantity to deliver per customer
s_i	– service time required per customer
$[a_i, b_i]$	– service time interval per customer
k	– a vehicle; $k \in K$; K – set of vehicles
o^k	– origin depot; $o^k \in V_D$
τ^k	– the time when a vehicle leaves the origin depot
f^k	– the fixed cost of using a vehicle
Q^k	– capacity of a vehicle
D^k	– distance limit a vehicle
$\widehat{D}^k$	– driving time limit a vehicle
W^k	– working time limit of a vehicle
r^k	– return to the depot of a vehicle
d_{ij}	– the distance between customer i and j
$\hat{d}_{ij}$	– driving time between customer i and j
c_{ij}^k	– the cost of the trip from customer i to j
Ω	– binary variable; 1 if vehicles have to return to deport at the end, 0 otherwise
Θ	– binary, variable; 1 if the time window is hard, 0 otherwise
β	– binary variable; 1 if there is a backhaul constraint, 0 otherwise

Decision Variables

x_{ij}^k – binary variable; 1 if vehicle k traverses from customer i to j, 0 otherwise
y_i^k – binary variable; 1 if vehicle k serves customer i, 0 otherwise
w_{ij}^k – pickup amount of vehicle k from customer i to j
z_{ij}^k – the delivery amount of vehicle k from customer i to j
t_i^k – time of a vehicle to arrive at customer i
v_i – late time of arrival at customer i

Constraints Explanation

Equation (22) represents the maximization of profit minus the cost of travel, the fixed cost of vehicles, and the penalty of being late.

Equation (23) and (24) represents the constraints that force vehicles to visit each customer once and exclude customers that don't need to be served.

Equation (25) represents a weak form of the well-known flow conservation constraints.

Equation (26) ensure the possibility of a connection between the depot of the vehicle k and the customers visited by this vehicle.

Equation (27) determines if the vehicle should return or not to the depot.

Equation (28) says that a vehicle cannot be used more than one time.

Equation (29) represents the backhaul constraints.

Equations (30) and (31) verify that the amount of pick-up carried by the vehicle is equal to the amount of pick-up required by the customer.

Equations (32) and (33) represent the same as equations (30) and (31) but for delivery items.

Equation (34) (35) (36) defines the time window constraint. The first equation in equation 34 explains that the addition of the time of arrival at customer i, the driving duration, the service time minus the working time limit should be less than the arrival time at customer j. The second equation (35) makes sure that the arrival time is between the beginning of the time interval of the customer and the end of the time interval minus the service time plus the delay time allowed if it is a soft time window. The last equation of Equation (37) determines if the time window is soft or hard.

Equations (37) and (38) determine the time when the vehicle should start working and ensure that it goes back to the depot on time if it must.

Equation (39) prevents the vehicle from having more objects to deliver and collect than its capacity allows.

Equation (40) set the distance limit for a vehicle k.

Equation (41) set the driving time limit for a vehicle k.

Equation (42) set the working time limit for a vehicle k.

Equation (43) to (47) just shows the values that the variables can take.

5.2.4 Results and Interpretation

4127 LNS iterations are carried out by the solver. A total net profit of 6515.27 NOK is calculated after all the demands of the customers were fulfilled. The net profit of each vehicle from each depot along with the individual profit collection, distance traveled, driving, arrival, and departure time is shown in Table 5-6 to Table 5-14 and the route from the depot to the customer is shown in Figure 5-5.

Table 5-6: Work Description of Vehicle V1

Vehicle:	**V1**	**Stops:**	**2**	**Net profit:**	**1353,87**	
Stop count	**Location Name**	**Distance travelled**	**Driving time**	**Arrival time**	**Departure time**	**Profit collected**
0	Depot A	0.00	0:00		07:00	
1	Customer 3	7.22	0:11	07:11	14:11	2500
2	Depot A	14.61	0:22	14:22		

Table 5-7: Work Description of Vehicle V2

Vehicle:	**V2**	**Stops:**	**3**	**Net profit:**	**1453.19**	
Stop count	**Location Name**	**Distance travelled**	**Driving time**	**Arrival time**	**Departure time**	**Profit collected**
0	Depot A	0.00	0:00		07:00	
1	Customer 5	2.33	0:06	07:06	11:06	1250
2	Customer 6	2.33	0:06	11:06	14:36	1250
3	Depot A	4.68	0:11	15:05		

Table 5-8: Work Description of Vehicle V3

Vehicle:	**V3**	**Stops:**	**2**	**Net profit:**	**103.87**	
Stop count	**Location Name**	**Distance travelled**	**Driving time**	**Arrival time**	**Departure time**	**Profit collected**
0	Depot A	0.00	0:00		07:00	
1	Customer 4	7.22	0:11	07:11	10:41	1250
2	Depot A	14.61	0:22	10:52		

Table 5-9: Work Description of Vehicle V4

Vehicle:	V4	Stops:	3	Net profit:	478.67	
Stop count	**Location Name**	**Distance travelled**	**Driving time**	**Arrival time**	**Departure time**	**Profit collected**
0	Depot B	0.00	0:00		07:00	
1	Customer 10	10.81	0:19	7:19	10:49	500
2	Customer 2	17.87	0:32	11:02	14:32	1250
3	Depot B	27.13	0:49	14:49		

Table 5-10: Work Description of Vehicle V5

Vehicle:	V5	Stops:	-	Net profit:	-	
Stop count	**Location Name**	**Distance travelled**	**Driving time**	**Arrival time**	**Departure time**	**Profit collected**
-	-	-	-		-	-

Table 5-11: Work Description of Vehicle V6

Vehicle:	V6	Stops:	-	Net profit:	-	
Stop count	**Location Name**	**Distance travelled**	**Driving time**	**Arrival time**	**Departure time**	**Profit collected**
-	-	-	-		-	-

Table 5-12: Work Description of Vehicle V7

Vehicle:	V7	Stops:	2	Net profit:	1394.00	
Stop count	**Location Name**	**Distance travelled**	**Driving time**	**Arrival time**	**Departure time**	**Profit collected**
0	Depot C	0.00	0:00		07:00	
1	Customer 1	5.28	0:10	07:10	14:10	2500
2	Depot C	10.60	0:21	14:21		

Table 5-13: Work Description of Vehicle V8

Vehicle:	V8	Stops:	3	Net profit:	369.00	
Stop count	**Location Name**	**Distance travelled**	**Driving time**	**Arrival time**	**Departure time**	**Profit collected**
0	Depot C	0.00	0:00		07:00	
1	Customer 7	6.53	0:11	07:11	09:11	750
2	Customer 8	6.53	0:11	09:11	14:00	750
3	Depot C	13.10	0:22	14:11		

Table 5-14: Work Description of Vehicle V9

Vehicle:	V9	Stops:	2	Net profit:	1362.67	
Stop count	Location Name	Distance travelled	Driving time	Arrival time	Departure time	Profit collected
0	Depot C	0.00	0:00		07:00	
1	Customer 9	6.89	0:13	07:13	14:13	2500
2	Depot C	13.73	0:26	14:26		

Only a single customer with higher demands (500 kWh) is served by the one vehicle for e.g., Customer 1 is served by V7 whereas, two customers with low demands (250 kWh) are served by one vehicle for e.g., Customers 5 and 6 are served by vehicles V2 which shows the optimal use of vehicles as per the demand of the customers. The algorithm can also serve for the pickup and delivery cases which is not considered in this case study because of time limitations. If a situation arises where it is required to pick up the detachable mobile battery containers and deliver it to a construction site and again go for the next pick-up and delivery it can also be performed.

Typical 07:00 – 15:00 working hours are considered for this case study, so all the work starts at 7 AM in the morning and ends at 3 PM in the afternoon. All the customers have their specific working times as stated in Table 5-3. The time at which the work starts is named 'Time Window Start' and the time at which the work ends is named 'Time Window End'. The vehicles leave their corresponding depot start 7:00, serve the customers within their time windows, and return to their depot at the end of the service. All the customers must be visited so, it's a hard constraint. The driving time is also mentioned in Table 5-6 to Table 5-14 which is calculated by Bing's driving time and distances considering the average vehicle to be 70 kilometers per hour.

To clarify the arrival time, departure time, driving time, distance traveled, and the profit collected let us look at Table 5-13. Vehicle V8 leaves depot C at 07:00, arrives at customer 7 at 07:11 after driving for 11 minutes covering 6.53 km, serves 2 hours there, and leaves at 09:11. It arrives at customer 8 at 09:11 because both the customers (machines) are located in the same places (construction site) so it doesn't take any time. It serves customers 8 for 5 hours and leaves at 14:00 and arrives back at depot C after driving for 11 minutes covering 6.57 km. The total distance traveled for vehicle V8 is 13.10 km and the total driving time is 22 minutes. Profit is earlier considered to be fives time the demand so, for 150 kWh the profit is 150*5=750. Vehicle V8 collects NOK 750 from customer 7 and NOK 750 from customer 8. Finally, the net profit is calculated by subtracting fixed cost (NOK 1000) and cost per unit distance (Nok 10 * 13.10 = 131) from the profit collected which is 1500 – 1000 – 131 = 369.

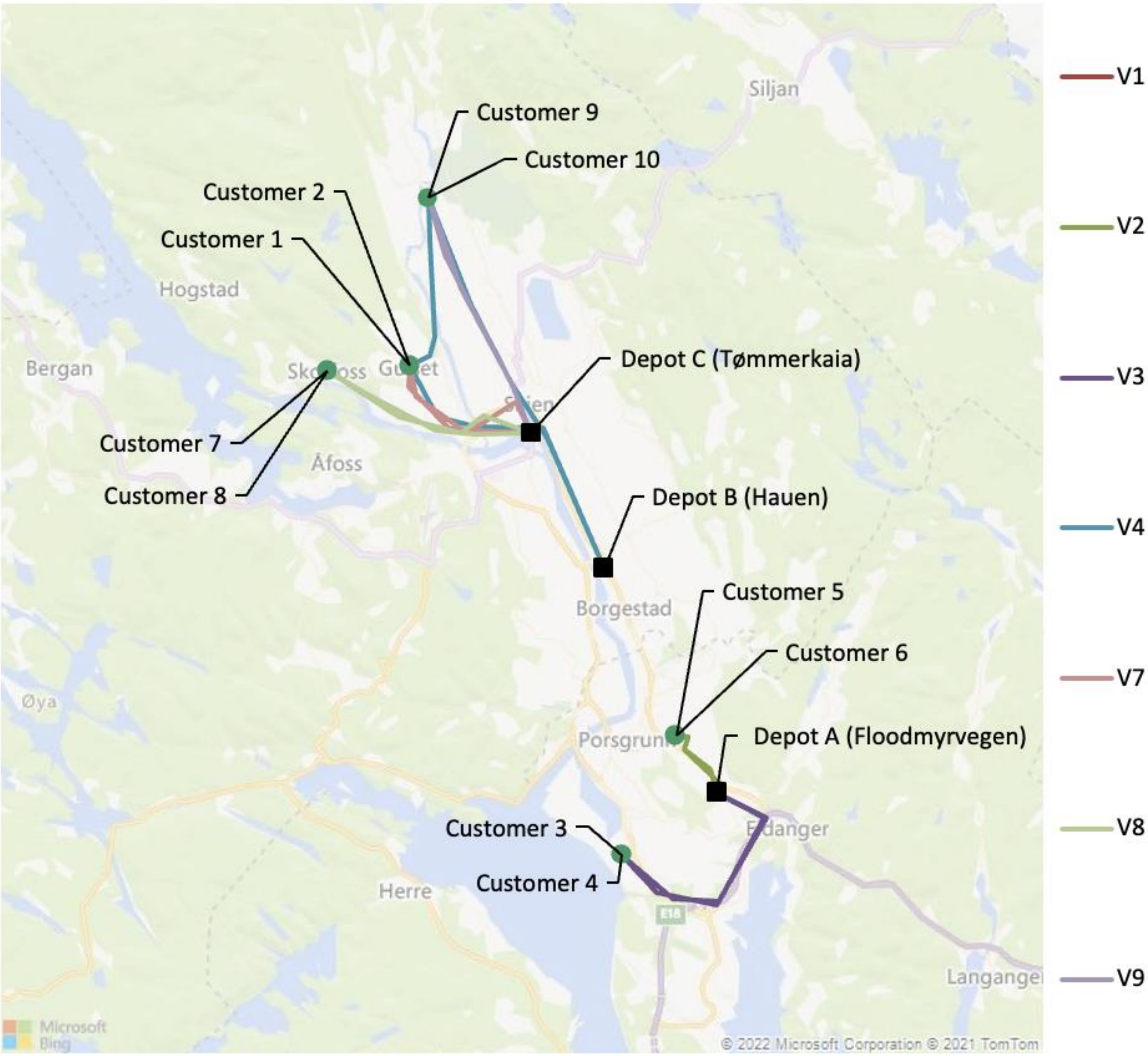

Figure 5-5: Route from Depots to Customers

Figure 5-5 shows the route of vehicles V1, V2, V3, V4, V7, V8, and V9 from their corresponding depots to customers and back. V1-V9 in the figure denotes all the nine vehicles belonging to their respective depots as mentioned in Table 5-4. Depot A, B, C, and Customer 1-10 in the figure represents all the depots and the customers as mentioned in Table 5-3. A total of nine vehicles, three in each depot, are made available out of which only 7 vehicles are scheduled by the algorithm, two vehicles V5 and V6 are not scheduled since all the demands are optimally fulfilled by the remaining 7 vehicles i.e., V1, V2, V3, V4, V7, V8, V9.

Table 5-15: Customer served by the corresponding vehicle

Vehicles	Customer Served
V1	Customer 3
V2	Customer 5 Customer 6
V3	Customer 4
V4	Customer 10 Customer 2
V5	×
V6	×
V7	Customer 1
V8	Customer 7 Customer 8
V9	Customer 9

The list of vehicles and the customers it serves are mentioned in Table 5-15. The algorithm specifically chooses not to schedule vehicles V5 and V6 because all the customers in the Porsgrunn area are served by all the vehicles (V7, V8, V9) in the nearest depot (Depot A) and all the customers in the Skien area are served by all the vehicles (V1, V2, V3) in the nearest depot (Depot C) but there is a need of an extra vehicle to fully serve all the Skien customers hence the algorithm uses just one vehicle (V4) from depot B.

6 Results and Discussion

In the first case study, a distributor aims to send a vehicle carrying battery containers to construction sites, where the need of each machine must be met by one battery container, container capacity for any vehicle must not be exceeded, and overall distance must be minimized. There are typically four categories of equipment, each with its own set of energy requirements for the rated machine. It is important to compute the total distance traveled, total energy demand, and the total number of mobile battery containers.

After successfully applying the problems in the Microsoft Excel solver, it is established that fourteen cars are necessary to meet all of the demands, as well as the path of the vehicle from the depot to the different customers. All of the criteria are met by exactly one vehicle, with a minimal number of vehicles required to meet all of the energy demands. If the vehicle arrives at the place with the lowest demand, it fulfills it before proceeding to the next customer and finally returning to the depot. If the vehicle arrives at a point with a higher need, it meets the demand and searches for the next demand; if it can be supplied, it serves it; if not, it returns to the starting depot.

As in the pilot study, the real-world problem is simplified and limited to a standard planning cycle in the second case study. The primary tasks in this study are to first schedule the vehicle to be used by customers at the construction site and recharged at the depot, and then to calculate the total cost associated with it based on the shortest distance from the depot to the construction site. In five construction sites, there are three depots and 10 customers. The depots' and customers' latitude and longitude, distances between them, driving time between them, operating time, delivery quantities, and profit earned are all computed. In each of the three depots, there are three identical vehicles, their fixed cost and cost per unit distance are considered which serve the energy need of all the customers.

After four thousand iterations of the large-neighborhood search algorithm, a total net profit is calculated after all the demands of the customers were fulfilled. The net profit of each vehicle from each depot along with the individual profit collection, distance traveled, driving, arrival, and departure time is also calculated by the algorithms. Only a single customer with higher demands is served by the one vehicle whereas two customers with low demands are served by one vehicle showing the optimal use of vehicles as per the demand of the customers. The algorithm can also be used for pickup and delivery instances, which are not covered in this case study due to time limitation. If the need arises to pick up the detachable mobile battery containers, deliver them to a construction site, and then return for the next pick-up and delivery, this can be done. For this case study, typical working hours are taken into account, so all work begins at a predetermined time in the morning and concludes at a fixed time in the afternoon. Every customer has a specified working time labeled 'Time Window Start' and 'Time Window End.' The vehicles leave their corresponding depot at their fixed time, serve the customers within their time windows, and return to their depot at the end of the service. All the customers must be visited so, it’s a hard constraint. A total of nine cars are made available, three in each depot, of which only seven are scheduled by the algorithm, and two are not planned because all requests are ideally met by the remaining seven vehicles.

7 Conclusions

The fossil-free construction site often employs bio-fueled construction equipment that produces other pollutants such as particulate matter and nitrogen oxide but is not carbon neutral, meaning that fossil-free does not imply zero emissions. In Norway, a trial project to provide electric energy to building activities in areas where access to the power grid is not available is now being evaluated. This thesis proposes a general mobile battery charging scheduling problem that involves using mobile battery containers to charge batteries at a location where the grid has adequate capacity and then driving the batteries from the charging station to relevant construction sites that use battery-powered construction machines. The optimization model is formulated as a MILP, with objective functions, constraints, and other important parameters, and it is then solved using the Microsoft Excel solver. Finally, two case studies are provided which help to understand the practical scenarios and challenges that come along with the solution.

Although there has been no specific previous research effort on the electric vehicle scheduling problem for emission-free construction sites, the core idea behind the traveling salesman problem, vehicle routing problem, vehicle routing problem with time window, vehicle scheduling problem, and battery electric transit vehicle scheduling problem has been useful for problem formulation. A scenario in which a construction company intends to replace all conventional diesel equipment with electric counterparts in an emission-free construction site is investigated for a general formulation of the problem. The main tasks in this study are to schedule the recharge of the mobile battery container and to identify the number of charging stations and chargers required.

Mixed-integer linear programming, mixed-integer non-linear programming, genetic algorithms, and large neighborhood search algorithms are investigated to address these formulations. These procedures are carried out using the MATLAB optimization toolbox and the Microsoft Excel solver. It can be concluded from these studies that noncommercial MILP software products, in general, cannot match the speed or reliability of their commercial counterparts, but they can be a viable alternative for customers who cannot afford the more expensive commercial alternatives. Open-source software tools can also be more expandable and easier to adapt for certain applications than commercial software tools, whose versatility may be limited by the user interface.

Two study cases are developed: one is a simple optimization problem to comprehend the concept of vehicle routing problems, and the other is a more difficult optimization scheduling problem to address a real-world scenario. These are constructed using mixed-integer linear programming and solved in an Excel solver using the large neighborhood search algorithm. The outcomes of these cases were satisfactory nonetheless it could have been more polished, and added more features, and constraints if there were no time limitations. Future work might cover a more complex study scenario without limiting the variables, it might as well cover the possibilities of partial charging and discharging, fuel usage and CO_2 emissions, and an algorithm that can present an alternative option when one or more limitations are not met.

References

[1] A. Berger, E. Thapa, H. R. Mohajeri, "Voltage Stability Simulation of High- capacity Charging Platform for Emission Free Construction Sites," Universitetet i Sørøst-Norge, Norway, 2021.

[2] M. Jünger, G. Reinelt, and G. RinaMi, "The Traveling Salesman Problem," 1995.

[3] "Travelling Salesman Problem," *Wikipedia*. 2022. [Online]. Available: https://en.wikipedia.org/wiki/Travelling_salesman_problem

[4] P. L. Aga, C. M. H. Kuijpers, R. H. Murga, I. Inza, and S. Dizdarevic, "Genetic Algorithms for the Travelling Salesman Problem: A Review of Representations and Operators," 1999.

[5] J. Robinson, "On the Hamiltonian game (a traveling salesman problem)," presented at the RAND PROJECT AIR FORCE ARLINGTON VA, 1949.

[6] P. Toth and D. Vigo, Eds., *The Vehicle Routing Problem*. Philadelphia: Society for Industrial and Applied Mathematics, 2002.

[7] Bodin, Lawrence, Bruce Golden, Arjang Assad, and Michael Ball, "Routing and Scheduling of Vehicles and Crews: The State of the Art." Computers and Operations Research 10, no. 2, 1983.

[8] X. Gan, Y. Wang, S. Li, and B. Niu, "Vehicle Routing Problem with Time Windows and Simultaneous Delivery and Pick-Up Service Based on MCPSO," *Math. Probl. Eng.*, vol. 2012, pp. 1–11, 2012, doi: 10.1155/2012/104279.

[9] L. Zhen, C. Ma, K. Wang, L. Xiao, and W. Zhang, "Multi-depot multi-trip vehicle routing problem with time windows and release dates," *Transp. Res. Part E Logist. Transp. Rev.*, vol. 135, p. 101866, Mar. 2020, doi: 10.1016/j.tre.2020.101866.

[10] S. Thangiah, "Vehicle Routing with Time Windows using Genetic Algorithms," in *The Practical Handbook of Genetic Algorithms*, L. Chambers, Ed. CRC Press, 1995. doi: 10.1201/9781420050073.ch11.

[11] D. Zhang, X. Wang, S. Li, N. Ni, and Z. Zhang, "Joint optimization of green vehicle scheduling and routing problem with time-varying speeds," *PLOS ONE*, vol. 13, no. 2, p. e0192000, Feb. 2018, doi: 10.1371/journal.pone.0192000.

[12] T. Liu and A. (Avi) Ceder, "Battery-electric transit vehicle scheduling with optimal number of stationary chargers," *Transp. Res. Part C Emerg. Technol.*, vol. 114, pp. 118–139, May 2020, doi: 10.1016/j.trc.2020.02.009.

[13] A. Stokes and L. Poger, "Electric Drive Buses. Research Report," *Adv. Energy*, p. 29, 2013.

[14] Y. Wang, Y. Huang, J. Xu, and N. Barclay, "Optimal recharging scheduling for urban electric buses: A case study in Davis," *Transp. Res. Part E Logist. Transp. Rev.*, vol. 100, pp. 115–132, Apr. 2017, doi: 10.1016/j.tre.2017.01.001.

[15] M. Wen, E. Linde, S. Ropke, P. Mirchandani, and A. Larsen, "An adaptive large neighborhood search heuristic for the Electric Vehicle Scheduling Problem," *Comput. Oper. Res.*, vol. 76, pp. 73–83, Dec. 2016, doi: 10.1016/j.cor.2016.06.013.

[16] N. Olsen, N. Kliewer, and L. Wolbeck, "A study on flow decomposition methods for scheduling of electric buses in public transport based on aggregated time–space network models," *Cent. Eur. J. Oper. Res.*, Aug. 2020, doi: 10.1007/s10100-020-00705-6.

[17] O. Sassi and A. Oulamara, "Electric vehicle scheduling and optimal charging problem: complexity, exact and heuristic approaches," *Int. J. Prod. Res.*, vol. 55, no. 2, pp. 519–535, Jan. 2017, doi: 10.1080/00207543.2016.1192695.

[18] L. Li, H. K. Lo, and F. Xiao, "Mixed bus fleet scheduling under range and refueling constraints," *Transp. Res. Part C Emerg. Technol.*, vol. 104, pp. 443–462, Jul. 2019, doi: 10.1016/j.trc.2019.05.009.

[19] J. P. Vielma, “Mixed Integer Linear Programming Formulation Techniques,” *SIAM Rev.*, vol. 57, no. 1, pp. 3–57, Jan. 2015, doi: 10.1137/130915303.

[20] C. H. Papadimitriou and K. Steiglitz, *Combinatorial optimization: algorithms and complexity*. Mineola, N.Y: Dover Publications, 1998.

[21] J. K. Karlof, Ed., *Integer programming: theory and practice*. Boca Raton, FL: Taylor&Francis, 2006.

[22] M. Padberg and G. Rinaldi, “A Branch-and-Cut Algorithm for the Resolution of Large-Scale Symmetric Traveling Salesman Problems,” *SIAM Rev.*, vol. 33, no. 1, pp. 60–100, Mar. 1991, doi: 10.1137/1033004.

[23] J. E. Mitchell, “Branch-and-Cut Algorithms for Combinatorial Optimization Problems,” *Math. Sci. Rensselaer Polytech. Inst. Troy NY USA*, 1999.

[24] T. Achterberg, T. Koch, and A. Martin, “Branching rules revisited,” *Oper. Res. Lett.*, vol. 33, no. 1, pp. 42–54, Jan. 2005, doi: 10.1016/j.orl.2004.04.002.

[25] D. Feillet, “A tutorial on column generation and branch-and-price for vehicle routing problems,” *4OR*, vol. 8, no. 4, pp. 407–424, Dec. 2010, doi: 10.1007/s10288-010-0130-z.

[26] C. Barnhart, E. L. Johnson, G. L. Nemhauser, M. W. P. Savelsbergh, and P. H. Vance, “Branch-and-Price: Column Generation for Solving Huge Integer Programs,” *Oper. Res.*, vol. 46, no. 3, pp. 316–329, Jun. 1998, doi: 10.1287/opre.46.3.316.

[27] “Integer Programming,” *Wikipedia*. 2022. [Online]. Available: https://en.wikipedia.org/wiki/Integer_programming

[28] M. R. Bussieck and A. Pruessner, “Mixed-Integer Nonlinear Programming,” *Univ. Wien*, 2003.

[29] P. Belotti, C. Kirches, S. Leyffer, J. Linderoth, and J. Luedtke, “Mixed-Integer Nonlinear Optimization,” *Argonne Natl. Lab.*, 2012.

[30] Wang, D.W., Wang, J.W., Wang, H.F., Zhang, R.Y. and Guo, Z., “Intelligent Optimization Method,” *Beijing High Educ. Press*, 2006.

[31] D. Whitley, “A genetic algorithm tutorial,” *Stat. Comput.*, vol. 4, no. 2, Jun. 1994, doi: 10.1007/BF00175354.

[32] Z. Cheng, “Optimal offshore supply vessel planning: A case study of a Chinese offshore oil and gas production area.,” University of Stavanger, Norway, 2013.

[33] M. Mitchell, *An introduction to genetic algorithms*. Cambridge, Mass.: MIT Press, 1996. Accessed: May 04, 2022. [Online]. Available: https://search.ebscohost.com/login.aspx?direct=true&scope=site&db=nlebk&db=nlabk&AN=1337

[34] S. Katoch, S. S. Chauhan, and V. Kumar, “A review on genetic algorithm: past, present, and future,” *Multimed. Tools Appl.*, vol. 80, no. 5, pp. 8091–8126, Feb. 2021, doi: 10.1007/s11042-020-10139-6.

[35] S. P, “Using constraint programming and local search methods to solve ve- hicle routing problems,” in *Princi- ples and Practice of Constraint Programming*, 1998, pp. 417–431.

[36] D. Pisinger and S. Ropke, “Large Neighborhood Search,” *Handb. Metaheuristics*, p. 23, 2010.

[37] S. Ropke and D. Pisinger, “An Adaptive Large Neighborhood Search Heuristic for the Pickup and Delivery Problem with Time Windows,” *Transp. Sci.*, vol. 40, no. 4, pp. 455–472, Nov. 2006, doi: 10.1287/trsc.1050.0135.

[38] MATLAB, “Optimization Toolbox User’s Guide,” *MathWorks Inc*, Sep. 2021.

[39] Frontline Solvers, “Premium Solver Platform For Mac, User Guide,” *Frontline Syst. Inc*, vol. 10.5, 2011.

[40] G. B. Dantzig and J. H. Ramser, “The Truck Dispatching Problem,” *Manag. Sci.*, vol. 6, no. 1, pp. 80–91, Oct. 1959, doi: 10.1287/mnsc.6.1.80.

[41] G. Erdoğan, “An open source Spreadsheet Solver for Vehicle Routing Problems,” *Comput. Oper. Res.*, vol. 84, pp. 62–72, Aug. 2017, doi: 10.1016/j.cor.2017.02.022.

Appendix

University of
South-Eastern Norway

Faculty of Technology, Natural Sciences and Maritime Sciences, Campus Porsgrunn

FMH606 Master's Thesis

Title: Optimal scheduling for mobile battery charging systems

USN supervisor: Carlos F. Pfeiffer (Porsgrunn); co-supervisor: Thomas Øyvang

External partner: Skagerak Energi

Task background:
A pilot project to provide electric energy to construction activities in parts where access to the power grid is not possible is presently under evaluation in Norway. The idea is to use mobile battery containers, charge batteries at a location where the grid has good capacity, and then drive the batteries from the charging station to the relevant construction sites that use battery-powered construction machines. When the batteries are discharged, the empty batteries are driven to the nearest charging station, and new, fully charged batteries are driven to the construction site.

Task description:
The thesis objective is to analyse the optimal scheduling problem considering several mobile battery containers, several plausible charging stations, and several construction sites with different needs. The scheduling should determine the important variables and constraints to consider, including constraints in charging capacity, charging times, possible delays, and construction project deadlines among others.

The main tasks are:

- Literature review on similar scheduling problems.
- Propose several realistic case scenarios for the scheduling problem (different number of batteries, charging stations with different capacities, different locations of construction sites, different deadlines constraints, costs, etc.). These scenarios should consider:
    - Construction operations that can utilize mobile batteries
    - Data from construction sites on electricity consumption, available hours for construction battery operated machinery, etc.
    - Emissions associated with transporting of batteries
    - Other factors to be discussed with the external partner.
- Build detailed optimization model including objective functions, constraints and important parameters to consider.
- Selection of adequate tools and algorithms to solve the optimization problem (for example, non-linear mixed integer programming solvers, genetic algorithms solvers, etc.)
- Solution and analysis of results of the proposed cases scenarios.
- Complete and deliver the master thesis report.
- Prepare and deliver a thesis presentation.