Faculty of Engineering

University Duisburg-Essen

Prof. Dr. Ing. Norbert Fuhr

Prof. Mohammad Amine Chatti

Vehicle routing with user-centred flexible delivery time window: A Case Study

A thesis submitted in
partial fulfilment of the
requirements for the degree of

Bachelor of Science
in
Computer Engineering (Software Engineering)

by

Azahar Hossain
Matriculation Number: 3056371

Submission Date:
12.01.2023

Acknowledgement

A warm felicitation goes for me to acknowledge the desirably encouraging individuals, praising, assisting, and believing me in the tasks of “Vehicle routing with user-centred flexible delivery time window: A Case Study” that I have analysed through my thesis.

First, I would like to take the opportunity to thank Prof. Dr Nobert Fuhr, my thesis’s principal supervisor, who has not only guided me through the process of choosing the appropriate topic but also provided valuable critiques. He has confronted me on several occasions to stimulate me to efficiently implement exposure on this thesis and to conclude it promptly according to my plan.

My special thanks go to Mr Marco Janc, the Chief Technical Officer and co-founder of Deloma UG, for his cordial trust and for immediately agreeing to guide me. This thesis would not have been created without the dedicated support of the company Deloma UG. I particularly express my honest and sincerest gratitude to Mr Janc, who has assisted me all the way. I want to thank him for sharing his technical intelligence and years of experience which have helped me to understand some crucial inputs at critical stages in my thesis.

I thank Mr Ibrahim Khalil, an employee of Deloma, who despite his busy schedule, gave immense assistance and information to write my thesis slickly. I would also like to thank my colleagues for their open and friendly nature and for keeping a good working atmosphere and support me whenever needed.

I am grateful to Mr Arif Ahmed, Muhammad Rashedul Islam, who is not directly related to my Bachelor Thesis but guided me through and motivated me a lot.

Last but not the least, I would like to thank my family, for providing me with unfailing support and continuous encouragement throughout my years of study and through the process of researching and writing this thesis. This accomplishment would not have been possible without them. Despite the loss of one of my dearest family members, the whole family encouraged me to move on in my academic steps.

Abstract

For the expanding E-commerce industry, vehicle routing is an intensive research focus. Using the open-source toolkit Jsprit, a case study on Vehicle routing was undertaken. The proposed Levelling Approach (LA) modifies user-centric time windows to achieve better-optimized routes. Users are prioritised, levelled, and set striction based on historical data and KPI. The study was conducted with about 5060 order instances. The effectiveness of LA is compared consequently against the existing tour data instances derived from the company named Deloma UG. Applying LA, the mean punctual orders increased internally for orders that scored high, as well as overall by about 37.1%. Besides, compared to existing data, the mean travel duration was reduced by about 1.3% and the mean travel distance by about 6.6%.

Keywords: *VRP, VRPTW, Jsprit, Open-Source Route Solvers, LA*

Confidentiality Clause

This study comprises several confidential data and information which are provided by the company “Deloma UG”. Those data and relative information may not be disclosed, published, or exhibited towards others in any circumstances, including the form of extracts, before or without the explicit permission of Deloma UG and the author. This research and the results are concluded to be available to the members of the Examination Board of University Duisburg-Essen solely for the purpose of assessment.

Table of Contents

List of Abbreviations

Abbr.	Explanation
LA	Levelling Approach
VRP	Vehicle Routing Problem
VRPTW	Vehicle Routing Problem with Time window
VRPFTW	Vehicle Routing Problem with Flexible Time window
SMEs[1]	**S**mall and **M**edium-sized **E**nterprise**s** Subdivisions: Micro enterprises (**<10** persons employed), Small enterprises (**10-49** persons employed) and Medium-sized enterprises (**50-249** persons employed)
KPI	**Key P**erformance **I**ndicator
ET	Early arrival Time
LA	Levelling Approach
LT	Late arrival Time
$\boldsymbol{s}$	Strictness score
$\boldsymbol{F}$	Flexibility constraint (time offset)
$\boldsymbol{R_{po}}$	The ratio of punctual orders per tour
$\boldsymbol{S_{dist}}$	Sum of travel distance per tour
$\boldsymbol{S_{dur}}$	Sum of travel duration per tour
$\boldsymbol{R_{po\_mean}}$	Mean value of punctual orders
$\boldsymbol{S_{dist\_mean}}$	Mean value of $\boldsymbol{S_{dist}}$
$\boldsymbol{S_{dur\_mean}}$	Mean value of $\boldsymbol{S_{dur}}$
C1	*Manual route planning* (Case 1)
C2	Case 2
C3	Case 3
Case 3V1	Case 3, version 1
Case 3V2	Case 3, version 2
Case 3V3	Case 3, version 3

[1] Source: https://ec.europa.eu/eurostat/statistics-explained/index.php?title=Glossary:Enterprise_size

List of Tables

List of Figures

List of Algorithms

List of Equations

1. Introduction

1.1 Motivation

In the past several decades, businesses have been able to adapt and even grow as a result of technological advancements in the digital world. The restrictions imposed by the Covid-19 outbreak prompted many businesses to either switch to the internet as a new sales channel or increase their efforts to sell their goods or services online, emphasizing the potential for digital technology to enhance the economic resilience of businesses. For example, in 2020, the Covid-19 pandemic drove 12% of EU businesses to begin or boost their online sales operations (Eurostat, 2022). In this digital age, consumers are more likely to shop online. Specifically, because of the COVID-19 pandemic, a noticeable increase in online shopping has been seen in recent years.

Over 38% of surveyed German customers completed online purchases in 2021, a nearly 9% increase from the previous year (Statista, 2022). Small and medium-sized Enterprises (SMEs) with fewer than 250 people are frequently described to represent the backbone of the European economy, and they have demonstrated reasonable resilience in 2020. There were 1 per cent more enterprises in 2019 than in 2018, with 23.4 million compared to 23.2 million. The increase was mainly in SMEs (Eurostat, 2022). For this increasing number of online purchases of the increasing number of enterprises, considering customer satisfaction factors, E-commerce companies need to focus on customer value to win the competitive market. So, they seek more effective and efficient marketing strategies that can influence customer purchase intention and business growth (Febrian & Fadly, 2021). Small and medium-sized enterprises (SMEs) frequently lack the internal skill sets to make their systems efficient (Pappalardo, 2021). According to a study on digital purchasing in the year 2020, due to delivery service problems, customer satisfaction dropped by about 6% compared to the previous year (Anon., 2020). Unni et al. (2015) conclude that satisfaction with the delivery time has an impact on overall customer satisfaction.

Customer satisfaction, a crucial factor in a company's expansion, occurs when the service meets or surpasses the customer's expectations (Khristianto, et al., 2012). This satisfaction differs from person to person; hence, loyalty influences repurchase intent. Nebojša, et al. (2019) concluded that numerous elements, including but not limited to, delivery time, price, website design, security, information quality, information availability, payment methods, e-quality of the service, product quality, range of products, provision of services, privacy, etc., can impact customer satisfaction. Although their result shows a considerably low impact on quality and time and a high impact on pricing. Considering the local shops and demographic conditions, other attributes, and restrictions of this research study (which are attached to the Annex, see List of other Restrictions) time, distance, sales volume, and a few other things (as shown in **Figure 1**) to be considered in this research paper. Additionally, service providers must meet or exceed certain requirements. Their satisfaction, such as increased sales volumes, the convenience of delivery locations that affect costs and profits, on-time payments, etc., are tied to the enhancements and growth in service quality. The quality of delivery service and the company's growth are dependent on sales resilience; therefore, it is essential to create a solution that enhances service and growth while limiting the risk of losing customers while having fewer employees.

Figure 1: Various factors of satisfaction

1.2 Problem description and statement

According to the E-Commerce Delivery Report by Parcel Perform and zenloop (Parcel Perform, zenloop, 2020), over 25 per cent of German and British customers are dissatisfied with the post-purchase experience of their deliveries, with the following as the leading causes of dissatisfaction: incorrectly advertised, delivery schedules, products not delivered, and bad customer service from logistics service providers. SMEs encounter challenges balancing customer satisfaction due to long delivery routes, distance, and expenses, whereas larger firms have a broader network and more trained professionals to design their own solutions.

This study examines the challenges faced by a German small and medium-sized enterprise (SME) named Deloma. Deloma UG[2] is a Microenterprise company, like other SMEs it also has below 250 employees, located in Krefeld Germany. For almost a decade it has been helping local shops, providing websites and other services (for instance: logistic solutions, software and hardware, tour planning, automated invoice services, payment methods integration, design, consultancy, advertisement, content etc.) who do not have their website. Currently, the corporation uses a manual method of site sequence selection to determine delivery routes for these orders. An example of how manual tour planning is done is shown in **Figure 2**. Through the tour planning module, delivery service providers (local shops) can plan tours using Drag and Drop feature. The User provided specific time windows (in which range the order can be delivered) are saved in the system. While planning an optimized route encounters problems considering both customer satisfaction (on-time delivery) and service provider requirements, including sales volume, travel distance, travel time, and capacity.

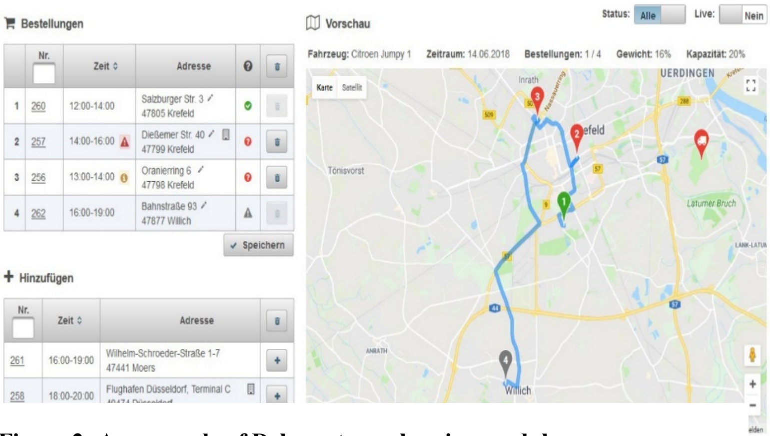

Figure 2: An example of Deloma tour planning module usage.

Graphic source: https://www.deloma.de/Agentur/wp/Logistik-Tourenplanung

Determining the best route to take while accounting for the considerations is known as the Vehicle Routing Problem (VRP). The theoretical background, methods to solve VRP, its variants, several tools and algorithms, and the consequential question that is raised to the above-mentioned problems, methods, and approaches are discussed in the next Chapters. The study focuses on the experimental outcomes of a real-life Case study by analysing historical data from three shops named Getränke Hax[3], Getränke Bob[4], and Getränke Bub[5]. These shops use the E-Commerce solution provided by Deloma UG.

---

[2] Deloma UG: www.deloma.de

[3] Getränke Hax Handels GmbH: A beverage delivery service in Essen / Mülheim. www.getraenke-hax.de

[4] Getränke Bob: A beverage delivery service in Dortmund. www.getraenkelieferant-dortmund.de

[5] Getränke Bub: A beverage delivery service in Duisburg, Düsseldorf, and Krefeld. www.getränke-bub.de

1.3 Thesis Outline

This thesis consists of several Chapters. Below, an overview of all the Chapters is discussed.

- **Chapter 1: Introduction.**

This Chapter introduces the basic concept of the topic and describes some problems some local shops in Germany are facing This Chapter provides an overview of the topic and explains the challenges that certain small businesses in Germany are encountering. A description of the problem's impetus and the pressing importance of its resolution are provided. There is also a summary of the entire paper.

- **Chapter 2: Theoretical Background.**

This Chapter provides the theoretical groundwork necessary for a comprehensive understanding of the topic at hand. The application of pertinent methodologies and technologies is examined. In addition to describing the problem's current state and currently available tools, this Chapter gives real-world illustrations.

- **Chapter 3: Aim and Research Questions**

The aims and objectives are presented in Chapter 3 along with the research questions that are set for this research. The hypothesis is formulated based on the theoretical basis and explained in this Chapter. Some research questions are prepared to answer.

- **Chapter 4: Methodology (Approach)**

Then Chapter 4 describes the research design, the research methodology (case study), data collection methods, data pre-processing, and some tools used.

- **Chapter 5: Case Study.**

This specific Chapter presents the case study inclusively the settings for the experiment, Apparatus, and Procedure. For three different cases, the application of the proposed approach, experimental procedures, and integrating tools are described.

- **Chapter 6: Limitations.**

Some research and technical limitations which have been experienced during the overall process are discussed in this portion. Additionally, a few more restrictions in this study are attached in the Annex.

- **Chapter 7: Conclusion and Future Works.**

In the final Chapter, the scientific base as well as the experiment to reflect on the approach, brief conclusions are highlighted and suggestions for future research are offered.

2. Theoretical Background

The theory that supports the research questions will be discussed in the following Chapter. It will explain the essential fundamentals and terminology needed to comprehend the significant parts of this thesis. Theoretical evolution and background of vehicle routing problems, solving algorithms, and solution approaches for different variants of VRP are described in Sections 2.1-2.3. Finally, a comprehensive summary is given in Section 2.4.

2.1 VRP

The Traveling Salesman Problem is a well-studied optimization problem in graph theory and operations research that can impede the multiple delivery process and cause monetary loss. In this optimization problem, TSP can be represented as a graph where all nodes are directly connected by edges or routes. The amount of weight assigned to each edge depends on the distance between each pair of nodes (Patel, 2022). The problem description can be given using a graph for example in **Figure 3**. Considering this directed graph G = (V, E) with a set of nodes V= {V0, V1, V2,V3,V4,V5,V6}, a set of Edges E ={e0, e1, e2, e3, e4, e5, e6.

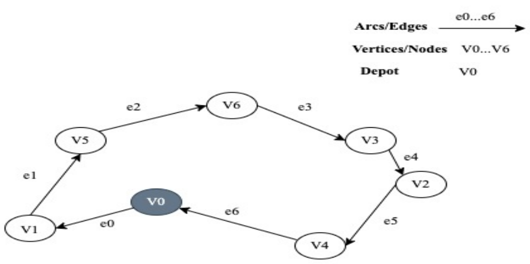

Figure 3: A directed graph showing edges and arcs.

Depot is denoted as V0. The weights of all e_i, (i = 0...6) represent the cost, between two nodes. The objective of this problem is to find the shortest path for this problem in other words to minimize total travel cost (in distance), while in Vehicle Routing Problems (VRP) additional constraints are to be considered. The classical version, with capacity constraint, is also known as the Capacity vehicle routing problem (CVRP) to distinguish it from other variants. TSP, VRP and its variants are well-known NP-hard[6] problems (Jan & H. G., 1981). The vehicle routing problem was first introduced by Dantzig and Ramser (1959), their main goal was to model a fleet of homogeneous trucks to serve a certain number of the gas station looking to minimize the total travel distance. A few years after Dantzig, an effective greedy heuristic was proposed by Clarke and Wright (1964). Following these two seminal papers, new methodological approaches for solving existing

[6] NP-hard(non-deterministic polynomial-time) : “A problem is NP-hard if an algorithm for solving it can be translated into one for solving any NP-problem (nondeterministic polynomial time) problem”.

vehicle routing problems were proposed. formalized the generic VRP as a linear optimization problem. (Toth, et al., pp. 1-3, 51-53). The authors Toth and Vigo (Toth & Vigo, 2002, pp. 2-6) as well as Nacima, et al. (2016) describe the typical characteristics of the routing and scheduling problems by considering their main components (road network, customers, depots, vehicles, and drivers), the different operational constraints that can be imposed on the construction of the routes, and the possible objectives to be achieved in the optimization process.

Road networks: Road networks can be described as directed or undirected graphs, where the arcs of the graph represent road Sections and the vertices correspond to the road junctions and the depot and customer locations.

Depot: The vehicles start usually from one or more depots and come back (sometimes do not). Depots are also located at the vertices of the road graph. Each depot is characterized by the number and types of vehicles associated with it and by the global amount of goods it can deal with.

Customers: geographically dispersed with specific requests or *demands*, to be visited and served. The fixed or flexible location of customers can be placed as vertices of the road graph. Customers can have specific time windows during which they need to be served.

A fleet of vehicles: A homogenous or heterogeneous fleet of vehicles can be associated with each depot and can be characterized by different load capacities, fixed and variable costs, compartment types and accessibility constraints such as a subset of edges that can be traversed by each vehicle.

The characteristics of the VRPs components, the nature of the demand and additional regulations (such as working periods during the day, number and duration of breaks, the maximum duration of driving periods, etc.) impose to comply with several operational and regulatory constraints.

Some constraints and variants of VRP:

Capacity constraints (**CVRP**): depots and vehicles may have limited capacities, which require the current load to not exceed the related limit and the customer demand must be satisfied.

***Priority constraints*:** In this extension of Vehicle Routing Problem with Backhauls (**VRPB**), also known as linehaul-backhaul, customers are divided into two subsets namely linehaul and backhaul. All the linehaul customers must be served before any backhaul customer may be served.

Time window constraints: customers must be served within both their time windows and the working periods of the drivers associated with the vehicle routes in which they are scheduled. Adding this time window constraint classifies the problem into variant Vehicle Routing Problem with Time Window (VRPTW). The authors Toth, et al. (2014, pp. 9-23) classify and compares different variants of VRPs depending on the *transportation request* such as VRP with pickup and delivery problems (VRPPD), Periodic VRP (PVRP), *intra-route constraints* such as Capacity VRP (CVRP), VRP with Multiple uses of vehicles (VRPM), *characteristics of fleets* such as VRP with multiple depots (MDVRP) and type of objectives. In **Figure 4**, some variants of VRPs are shown. In dynamic capacitated Vehicle Routing Problem (DCVRP) variants, information revealed over time consists of customers' locations and demands. Depending on travel duration and availability of vehicles the routing planning can be affected, thus rescheduling might be needed. Some other variants of the problem are Energy consumption and pollutant emission control in the routing, widely known as Green VRP (GVRP), and Split Delivery VRP (SDVRP). Tilk, et al. (2021) introduced the vehicle routing problem with delivery options (VRPDO) extending the Generalized Vehicle Routing Problem with Time Windows (GVRPTW). Alexander, et al. (2020) introduced an extension of VRP enhancing demand side known as VRP with flexible delivery locations (VRP-FL) as well as with additional time window constraints as VRP with flexible delivery locations (VRPTW-FL). In VRPTW-FL, each customer must be served at exactly one capacitated location among a set of multiple alternatives.

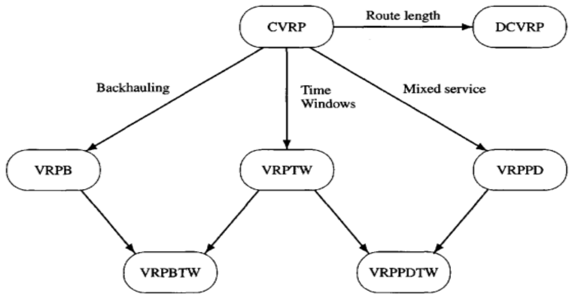

Figure 4:Variants of the VRP class and their interconnections.

Graphic source Toth, et al. (2014, S. 6)

Considering constraints and restrictions (see List of other Restrictions) from the company, this study fits more into Vehicle Routing Problem with Time Window (VRPTW).

2.2 VRPTW

The Vehicle Routing Problem with Time Windows (VRPTW) extends the Vehicle Routing Problem (VRP) where service at each customer must commence inside a time window (Desaulniers, et al., 2014, p. 119). Solomon (1987, p. 254) introduced the design and analysis of algorithms for vehicle routing and scheduling problems with time windows (VRPSTW). Time windows may be hard or soft, can be single or multiple, and can be fixed or flexible. Depot, employees, drivers, customers everyone can have fixed or flexible time windows. The time windows may be indicated as the latest time for delivery, for example. Problems with time windows occur naturally in business organizations with flexible time schedules. For private individuals usually, the time window is fixed but it can also be flexible (2014).

2.2.1 Hard time window

If a vehicle arrives at a customer's location too early, the driver has to wait until the customer is ready to be served. Waiting until the beginning of a time slot typically does not incur any additional costs.

However, this specific waiting time affects the next deliveries and other customers' satisfaction levels since they also expect a Just-In-Time (JIT) delivery. In **Figure 5**, delivery time is plotted against customer satisfaction level. The customer satisfaction level is high when delivery is on time. Security patrol service, bank deliveries, postal deliveries, industrial garbage collection, food delivery, school bus routing, and urban newspaper distribution are examples of services having specific time window issues. The hard time window must not be violated (Hokey, 1991, p. 179).

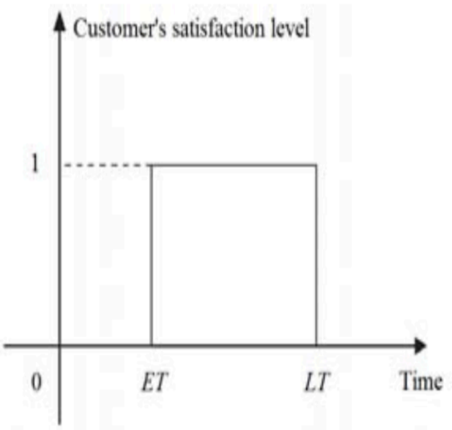

Figure 5: Customer satisfaction level in a hard-time window

Graphic source: (Tang & Huang, 2018, p. 362)

2.2.2 Soft time window

In the case of soft time windows, every time window can be violated barring a penalty cost which means it is the vehicle to start service at the customer before or after its time window (Jean-Francois, et al., 2002, p. 157). Among the soft time window problems, dial-a-ride problems constitute an important example (Hokey, 1991).

2.2.3 User-centred time windows

In a traditional VRPTW, a viable solution must meet all time windows, which is called a hard time window or for soft windows, the violation is allowed with a penalty cost. User-centred time windows are flexible time windows between hard and soft time windows. The customer's satisfaction level is determined by the actual arrival time. If the actual arrival time differs far from when the time window starts, then the driver has to wait if the time window is hard. In flexible time windows of customers, the customer's time window can be violated, thus in a sense, it can be considered as a soft time window. However, depending on the flexibility of the time window penalty might occur.

In this paper, the soft time window constraint is adopted. The flexibility of the time window as well as the penalty doesn't depend just on routing constraints but also on the customer's historical data and from a business perspective.

2.2.4 State of the art

Solomon (pp. 254-264) introduced several algorithms such as savings, insertion and sweep and concluded that in a real-life dial-a-ride environment, the insertion-based algorithm showed excellent performance for solving time window-constrained vehicle routing problems. Numerous variants and advancements have been introduced since the first solution approached. Some variants of VRPTW with two objectives namely minimizing the number of vehicles and the total distance (Nalepa & Miroslaw), variants with vehicle multi-trips planning, and multiple time windows of customers (Jacek, 2020). For the VRP with soft time constraints, Taillard, et al. (1997) suggested a tabu search-based metaheuristic. The issue of hard time window constraints can also be addressed with this method by severely punishing any delays (Jean-Francois, et al., 2002, p. 162).

Tang and Huang (2018) presented a new concept and application for solving cold chain vehicle routing problems with fuzzy time window constraints and two objectives. The results of their study indicate that their proposed technique has the potential to provide substantial cost savings and an acceptable customer satisfaction level.

To simplify the booking of time-window-based attended home deliveries, Köhler, et al. (2020) proposed a customer acceptance approach that allows some flexibility in the time window. Their approach evaluates delivery routes and decides if long or short delivery windows are practical for each new customer order. AllyouneedFresh, a Berlin-based online supermarket, is the source of their experimental data instances and customer demand characteristics.

The outcomes of these two studies suggest approaches for solving VRPTW with flexible time windows. In this paper, user-centred flexible time windows are focused on. Instead of passing the routing algorithm to decide which user's time windows can be violated and which not, a new methodological approach is introduced in Section 4.4.

2.3 Solution Approaches

When the number of customers in the problem is small and the optimal solution can be identified in a reasonable amount of time, exact methods are usually used. Since VRPTW is also an NP-hard problem, numerous heuristics and metaheuristics have been developed. This is because the precise algorithms required to fix the issue are too time-intensive for large applications (Paweł, et al., 2020). In contrast to simple heuristics, which strive to generate a single result, local search strategies and metaheuristics analyse several options.

This can be accomplished by developing a succession of solutions or by focusing on the population. Both simple and adaptable methods are used by commercial VRP software to quickly find viable and of good quality solutions. Simple heuristics are used in metaheuristics to create initial solutions. Two techniques can improve the solution quality of simple heuristics: the "**best of**" method and the "**randomization method**". The first method accepts the best of all possible solutions, and the second method randomizes the decisions of the original deterministic. (Nacima, et al., 2016, pp. 21-27)

2.3.1 Local Search heuristics

Most effective heuristics for the VRPTW revolve around the idea of a local search. The starting point for a local search is the initial solution, which is often found via a simple heuristic procedure. Then the solution space moves considering a subset of possible solutions, called Neighbourhood. Moving to another neighbourhood can cause cycling. It's necessary to take care to avoid cycling. There are several possible ways to avoid this, such as in Simulated Annealing (**SA**) this is prevented by selecting a solution x randomly from the solution subset. Other well-known local search algorithms include deterministic annealing (**DA**), Tabu Search (**TS**), Iterated local search (ILS), and variable neighbourhood search (Gilbert, et al., 2014).

2.3.2 Population-based heuristics

The population-based strategies are grounded on natural notions, such as the development of species and the behaviour of social insects when looking for food. These systems utilize a high-level guiding methodology based on various memory structures, such as neural networks, solution pools represented as chromosomes, and pheromone matrices, among others. Moreover, all known efficient VRP algorithms of this type rely on local search components to direct the search toward feasible solutions. This holds regardless of the problem's complexity. Consequently, the vast majority of population-based strategies mentioned in the VRP literature are hybrid by definition. Some well know population-based heuristics include: ant colony optimization (Marc, et al., 2004), Genetic algorithms, (John, 1992), and Neural Networks (J. J. & D. W., 1985) (Gilbert, et al., 2014, p. 92).

2.3.3 Simulated Annealing

The simulated annealing (SA) algorithm for combinatorial optimization was introduced by Kirkpatrick, et al. (1983) and was inspired by a physics phenomenon concerning the behaviour of atoms under temperature changes. The simulated annealing method transforms the annealing process into the minimization of an optimization problem's objective function, which is equivalent to the energy of a material, by introducing an imaginary temperature (Patrick, 2016). This process is a form of stochastic relaxation strategy that has its roots in statistical mechanics and is inspired by the annealing process for solids, in which a material is heated to a high temperature and then slowly cooled to cause the crystals to form in a low-energy shape.

In this perspective, SA may be viewed as an effort to free the fundamental hill-climbing dynamics from the grip of low-quality local maxima. The random transformation to perform on the incumbent solution to jump into the solution space in other words a particular neighbourhood. In the vehicle routing context, moves are often classical, such as relocating, exchange, and two-opt moves. SA guides the original local search method in a probabilistic way to escape local optimum and cycling (NEO Research Group, 2013).

2.3.4 Large neighbourhood search

In Large neighbourhood search (LNS) Part of the solution is destroyed before it is repaired. This combines elements of simulated annealing and threshold-accepting algorithms. To destroy some random solution, often greedy heuristics are used. For instance, in the CVRP, removing n customers at random from the routes and then re-insert them using the cheapest insertion heuristic. As a result, it improves the likelihood of locating high-quality locally optimum solutions, while simultaneously facilitating fast neighbourhood searching (Nacima, et al., 2016, pp. 67-68).

2.3.5 Ruin and Recreate principle

Incorporating a large neighbourhood search, Schrimpf et al. (2000) proposed the Ruin & Recreate principle. The researchers have examined their meta-heuristic on a wide range of Vehicle routing formulations, emphasizing its performance for problems involving complex optimization, discontinuous search spaces, and many constraints. In general, the Ruin & Recreate heuristic follows three steps:

Initial step: a plausible initial solution for a set of routes is built. Consideration is given to a present set of limitations for each vehicle; depending on VRP versions, these constraints may vary.

R & R step: multiple iterations of executing the Ruin & Recreate steps are carried out.

- **Ruin**: Some jobs are then removed from the existing job sets. To select which jobs to be eliminated a variety of methods such as Radial ruin, Random Ruin, Sequential Ruin, etc. In Radial Ruin, a random node c is selected from a set of nodes, removing c and its A-1 nearest neighbours while in Random Ruin removes A randomly selected node globally. A is less or equal to the product of a total number of nodes and a fractional number between 0 and 1. The nearest neighbours can be defined as Euclidean distance metrics in the vehicle routing context. In Sequential Ruin, succeeding nodes of selected A from a single randomly selected round trip.
- **Recreate:** In this step, re-inserting some or all the eliminated jobs into the schedules to find a new configuration to obtain a feasible solution once again with the best insertion technique. This heuristic yields after every iteration a feasible solution, but the quality is not assured to be optimal. A cost function based on

some constrained variables such as the distance or time driven by the vehicles can be defined to assure a better quality of the solution. Using cost matrices that show how much it will cost to travel between points along the route, the route's cost may be calculated based on the sequence in which procedures can be accomplished. The step consumes most of the computing time.

Threshold accepting step: from the iterated R& R steps several feasible solutions can be found. Based on the threshold acceptance algorithm like Simulated Annealing, chooses whether to preserve the old solution or go forward with the new one. Therefore, enabling a momentarily deteriorated objective to find better solutions in subsequent iterations (Schrimpf, et al., 2000).

In this paper, a vehicle routing toolkit is incorporated that uses the R&R principle. More about the tool and its usages is discussed later in Section 4.3.2

2.4 Summary

This Chapter discusses the general theory behind the Vehicle routing problem, some of its variants as well as solving strategies. Time-window-constrained variants with soft and hard time windows are discussed and state-of-the-art presented. The related heuristics and metaheuristics are also presented.

3. Aim and Research Questions

This paper presents the introduction of some VRP tools and algorithms to automate the routing task considering historical data of user orders and aims to evaluate the effectiveness of a methodological approach in different cases.

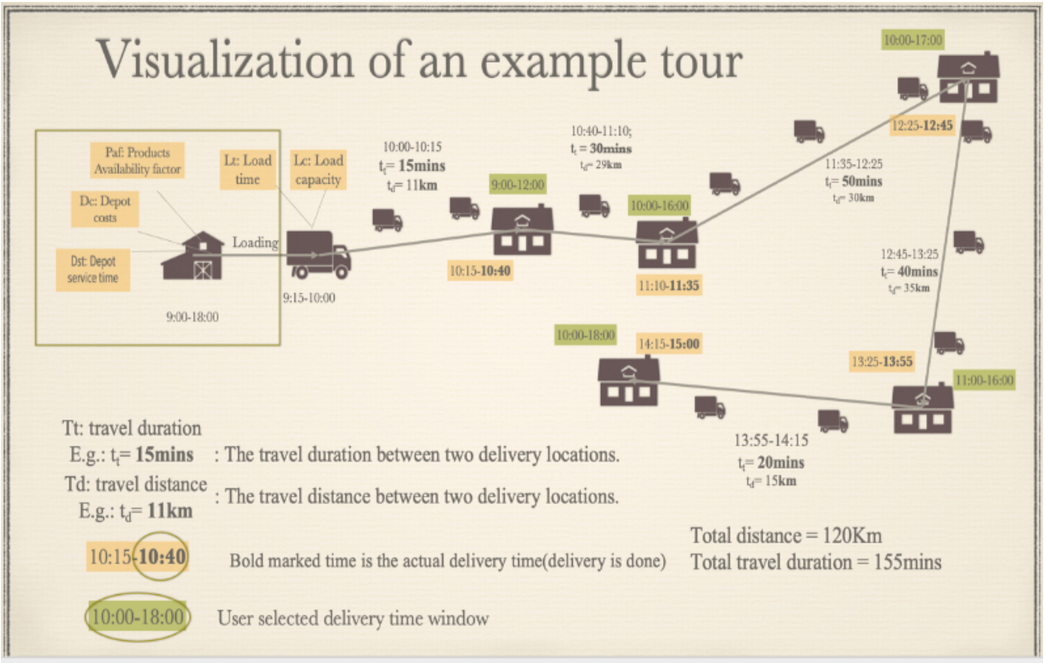

Figure 6: A dummy example of a tour to explain aim and objectives

The key objectives are evaluating the effectiveness of the VRP tool with an extended new approach applied, considering 1. decrease total costs (travel duration and distance) and 2. improve customer satisfaction (punctual order deliveries).

For a better understanding, consider the dummy example tour in **Figure 6**, the total travel duration is 155 minutes, and the total travel distance is 120 Kilometres. The values would change if the sequence of delivery location changes, for example from depot to location 1 is not the same as from location 1 to location 2 in the example tour.

To evaluate the effectiveness, an increment or decrement in punctual deliveries and a reduction of travel costs are focused on. However, satisfaction does not necessarily rely only on punctual delivery, yet as discussed earlier that delivery time could be a possible factor of satisfaction. To meet the goal few research questions are formulated, and hypotheses are stated in the following Sections.

3.1 Research questions

This study is focused on investigating and answering the following Research Questions:

1) How can the incorporation of a vehicle route optimization tool with historical data benefit Deloma system users (local shops)?
   a. How effective is automated route planning to reduce distance and time?
   b. How can Deloma system enhance customer satisfaction by improving the time flexibility of delivery service?

3.2 Hypotheses

H_0: If the fulfilments of the delivery time flexibility constraints (user-centred) for each Order are optimized, it doesn't affect the satisfaction level of customers (punctuality of deliveries), total travel duration, and travel distance at all.

H_1: If the fulfilments of the delivery time flexibility constraints (user-centred) for each Order are optimized, the satisfaction level of customers (punctuality of deliveries) will improve, and total travel duration and travel distance will decrease.

4. Methods and materials

This Chapter discusses the research approach and techniques for this bachelor's thesis. The Research design, methodology, and strategies are detailed here. This Chapter will also include methods for collecting data cleaning and pre-processing the collected data, and subsequently some data analysis techniques.

4.1 Research methodology (Case Study)

Case study research investigates complicated phenomena such as current events, significant challenges, and efforts to get a deeper understanding of them. This type of research typically focuses on contemporary issues rather than historical ones. Bounding a case study refers to what to include or exclude in the study. Using research questions, researchers determine the research subjects, specific geographical places, and a time limitation to study (Tricia, et al., 2012).

In this research, a case study is conducted. Identifying the aspects of the existing case the research questions are formulated in Section 3.1. As in the very beginning of the paper in Section 1.2 mentioned, this study is conducted with data from a Deloma database that is being used by E-commerce shops based in 3 different cities in Germany.

The time limitation is about 3 months. However, the preparation and workloads as well as the contribution to data collection that Deloma UG employees have provided such that the existing case can easily be identified, are excluded from the study time limitations. Besides, this study excludes multiple constraints in problem aspects such as the location radius of delivery constraints above 50 km, deliveries with multiple location demands, customers with multiple time windows, etc.

The study has been conducted to evaluate 3 different cases. Which are discussed in the next Chapter. To investigate these cases, data collection and data analysis are needed. The methods of data collection, pre-processing, and, if necessary, cleaning is outlined in the upcoming Section.

4.2 Research design

Researchers plans for the research implicitly or explicitly. According to Yin (2016, pp. 60-61), A research design is a logical plan to find a set of conclusions from a set of questions to be addressed. In this research, a set of questions are focused to be answered. To conclude what the findings are, a small workflow is planned.

Vehicle routing is a highly extensive research field. State-of-the-art and consistent theory adoption as well as solving it practically is a lengthy process. However, fitting the problem into a scientific research gap, considering the company's business perspective and the duration of the research period, this research is designed as shown in **Figure 7**.

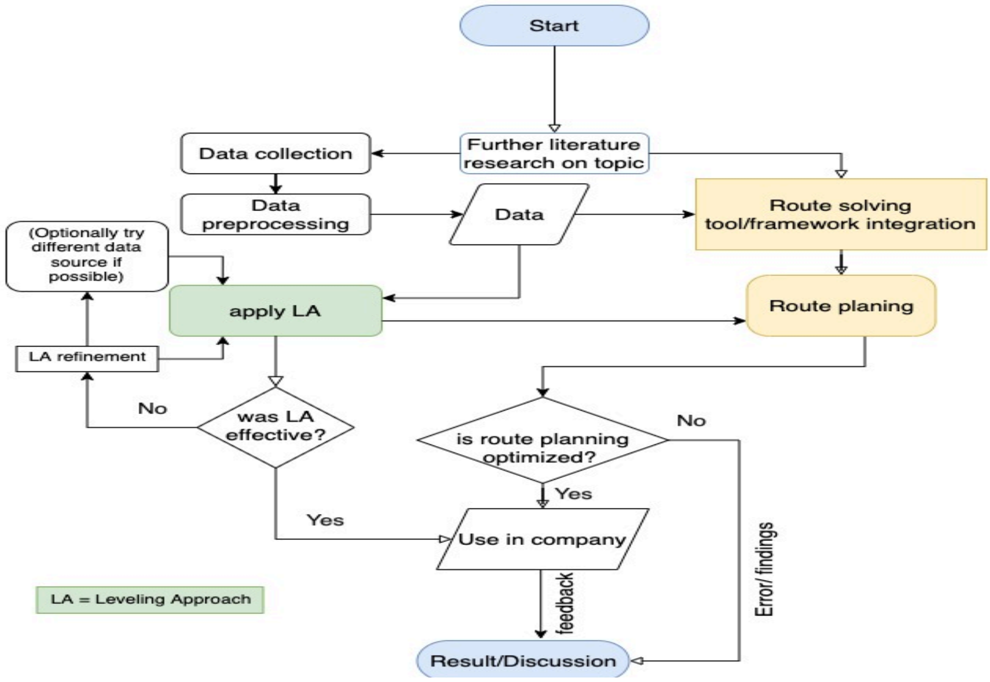

Figure 7: Research Design

Data collection, data pre-processing, and incorporation route solving framework as well as the new approach is proposed in the extension of using existing algorithms. A brief description of the input data and the pre-processing of the collected data is explained in the following Sections (see **5.1.1**).

Instead of proposing a new Model or theory to solve the Vehicle routing problem, this study is designed to evaluate the effectiveness of the existing tool in real-life data sets in different cases. From Data collection to data processing as well as the approach highlighted as LA in the design are discussed in the following Sections.

4.2.1 Data Collecting Method

Data can be collected in several ways. Yin (2016, pp. 153-160) lists "documentation", "archival records", "interviews", "direct observations", "participant observation", and "physical artifacts" as examples of commonly observed evidence sources from which data can be gathered. Among them, archival records can be several types of data sources such as service records for a specific given period, organizational records, maps, charts, survey data, or even files that are publicly available.

In this research, for the data collection native MySQL queries are used to collect the data from the company database as a structured, tabular data source. From a huge data set of historical deliveries, demographic and organizational data, user information, as well as numerical values that are highly required for answering the research questions, are collected. The pre-processing part of the dataset so that it can be used for analysis and evaluation is discussed in the upcoming Chapter (see Section 5.1.1).

4.3 Tools and frameworks

As this study aims to generate routes, a VRP solver library, "***Jsprit"*** with version "jsprit-core1.9.0-beta.9" is used. For API[7] requests and parsing the responses, other required libraries are used. After generating routes, the new data instances are saved in the case study database and the exported results are visualized using "Google Sheet". Google Distance Matrix API is used for the calculation of distance and duration. Both Jsprit and this API as well as a new proposed approach are discussed briefly in the next Sections.

4.3.1 Google Distance Matrix API

Google provides a variety of APIs such as "Direction API", "Places API", "Geocoding API", etc. Distance Matrix API is a web service API that determines how long it would take to get from one place to another given a matrix of starting and ending points. It is organized as rows that include values for duration and distance for each possible pair of origins and destinations. The API returns information on the optimal route from point A to point B. Passing parameters for various modes of transportation, requesting the API in varying measures such as in meters, seconds, miles, etc. helps to calculate the amount of time a trip will take in traffic. More details about the API are well documented on the Google Distance Matrix API Documentation page.

In this research, Google Distance Matrix API is used to calculate the travel duration and travel distance among delivery locations and depot locations.

---

[7] API: Application Programming Interface, works as a connector among source codes and third-party software. (IBM Cloud Education, 2020)

4.3.2 The Jsprit framework

Jsprit is an open-source toolkit project, which is currently maintained by the "Graphhopper"[8] developer team. According to the GitHub page of the project, 4 repositories are dependent on this toolkit who also aims to solve vehicle routing problems.

Being written in Java, it is straightforward to integrate using maven or by downloading the binary distributions. The toolkit is easy to use, portable, and versatile. It implements the "Ruin and Recreate (**R&R**)" principle (which has been discussed earlier in this paper in Section 2.3.5) in a way that is both modular and scalable. It can solve a broad variety of routing problems that are beyond the scope of conventional CVRP and the classical TSP, such as "Vehicle Routing Problem with Time Window (VRPTW)", "VRP with Pickup and Delivery (VRP-PD)", "Multiple Depot VRP", "VRP with backhauls", "VRP with a heterogeneous fleet", "Dial-a-Ride Problem (DARP)", etc. Adapting the "**best of**" (see in 2.3) the number of search strategies is kept lower such that it appeals to a simpler structure. On one hand, it becomes easier to check the constraints even for complex problems, on the other hand using "**the best of**" method the algorithm results comparably best of all solutions instead of just checking one.

For Radial Ruin (see more details in Section 2.3.5) is set as best 0.5 and the fraction 0.3, on the contrary in Random Ruin globally both best and fraction are set to 0.5 (Stefan, 2016), which is similar to the experimental settings of Schrimpf et, al. (2000). For the threshold acceptance, the number of iterations is set to 2000.

Usage of Jsprit in research, case studies, and projects:

Some other open-source projects such as "openrouteservice" use "Graphhopper" internally, Whereas Graphhopper solves the routing tasks using "Jsprit". "Open Door Logistics (**ODL**) Studio" ingrates Jsprit as a route solver directly. Many different types of studies have made use of the open-source solver. For instance, the effects of charging limits on a simulated fleet of autonomous delivery robots in Lyon, France are examined by combining Jsprit with the simulation program MATSim (Ayman, et al., 2022). Ricardo, et al. (2020) also integrate Jsprit with MATSim for a case study focusing on food retail distribution in Berlin, to solve vehicle routing problems with vehicle type-dependent range constrain. Villanueva (2020) has used ODL for a real-life waste collection study in

[8] "Graphhopper": An open-source routing Software by Graphhopper gmbH".

Stockholm, while Karkula, et al (2019) provides a comprehensive analysis of existing three open-source software namely VROOM, Google OR Tools, and Jsprit, for addressing time-window-capable vehicle routing problems.

In this study, initially Google OR tools were planned to use for finding routing solutions. Indeed, the test instance with minimal constraints was tested and resulted out to be a not feasible solution for vehicle and time constraints (see GitHub issue). As the research duration is limited, the author managed to incorporate an alternative tool, Jsprit is lightweight and widely used, so later in this study, Jsprit is used for solving the routing problem.

4.4 Levelling Approach (LA)

In this Section, a methodological approach is introduced to find a flexible time window based on historical data as input. To find a flexible time window, the strictness score ***s*** is to be found out, then an offset value ***o,*** as well as a ***flexibility Time-constraint f,*** is to define. Finally, the given time windows are adjusted accordingly (see "Algorithm 1").

Dependent variables such as ordered quantity and average punctuality score in the past can be derived from input parameters. Only a single time window is provided which is within a single-day range (*Between 00:01 and 23:59*). Users having multiple time windows such as 10:00 to 12:00 and 14:00 to 16:00 (for example having a lunch break in between) are not considered in

Algorithm 1. LEVELLING APPROACH (LA)

Input*: order data (o), with user historical data (u)*

Output*: order with updated time windows*

1. ***Initialization of variables****: assign zero to variables s, o and f*
2. *while (o and u don't have invalid input parameters) // find out the level*
3. *s ← calculate the strictness (s) score based on input parameters //average score of all strictness consideration*
4. *o ← calculate offset (o) based on s //* ***percentage***
5. *f ← calculate flexibility constraint (f) based on o //* ***time offset in minutes***
6. *adjust time window based on f value //****time window manipulation step***
7. *End*

this case study and can be established further research criteria in the future.

The priority of these dependent variables as well as the **offset** value to adjust the time window (***f***) can be changed depending on the implementation case. Levels or categories are for interpreting how strict or flexible the customer is based on the calculated score.

4.4.1 Score, offset and time flexibility constraint calculation

Based on the given input parameters order and user's historical data, for each dependent variable of the input parameters (such as sales volume of the order, order quantity, average punctuality score of the user in historical data, average order frequency of the user, etc.) a fractional value, so-called score S_i is set. A final score for **n** number of dependent variables is calculated using the following linear regression equation (see **Equation 1**).

Equation 1: Calculating the sum of weighted individual scores (*S_f*)

$$S_f = \sum_{i=0}^{n} S_i f_i$$

Here S_i refers to all individual scores and f_i to weight factors. The factors for score weight define how important the score is to be valued in the function. The target value is expected to be $\boldsymbol{S_f} \geq 0$. The sum of all factors ($\boldsymbol{f_i}$) can be found using the following equation:

Equation 2: Finding the sum of all factors (*N_f*)

$$N_f = \sum\nolimits_{s=1}^{n} f_s$$

Here, n is the number of total factors, the sum of these must not be 0. These factors, $f_1 ... f_n$ are constant fractional positive weighted values for setting the priority of the score values. Now, using both equations the final score can be obtained by using the following equation:

Equation 3: Calculating the final score (*s*)

$$\mathrm{s} = \bar{S} = \frac{S_f}{N_f}$$

The calculated strictness score ***s*** is a fraction percentage number ($0 \leq \mathbf{s} \leq 1$). The usage of these equations in this study can be found in the following Chapter**.** After calculating the score **s**, an offset ***o*** is to be determined using the following linear equation:

Equation 4: Finding the per cent offset (*o*)

$$o = ms + b$$

Here in **Equation 4**, ***s*** is the score, ***m*** is the slope and ***b*** is the offset-intercept. For both slope and offset-intercept values changes, the offset value increases or decreases. The offset function results in a value which denotes how much per cent of the actual time window difference should be manipulated.

Equation 5: Finding time flexibility offset (*f*)

$$f = o * t$$

Suppose a difference between time window ranges is ***t*** minutes, then one can obtain ***f*** *by* multiplying t with the offset per cent (see **Equation 5**). The value of *f* is a discrete number (minutes) which is to be added or subtracted from the given actual time window.

4.4.2 An Example of applying LA

An example of choosing levels or in other words the flexibility offset in minutes (***f)*** based on strictness score, is shown in **Table 1**, which categorizes customers into five categories, with their according to strictness *ranges*.

Table 1: An Example of LA parameter setting. *The tables show an example of manipulation of a given Time window. Assuming the scores are calculated and given for the given example. Based on the calculated score, for a given time window, offset o is calculated. Using* **Equation 5***, flexibility constraints f is calculated. New Time windows with a new earliest start ET' and latest start LT' are the manipulated time windows and results. Here for all example scores a constant time window is considered for better understanding.*

Level	*Score range* (%)	*S (%)*	*o (%)*	*f* (minutes)	*Manipulated TW*		Input Info	
					ET'	*LT'*		
Very flexible	0-20	0	15.0	18.0	09:42	12:18	Actual TW (range)	
		0.2	10.0	12.0	09:48	12:12	ET	LT
Flexible	21-40	0.3	7.5	9.0	09:51	12:09	10:00	12:00
		0.4	5.0	6.0	09:54	12:06		
Normal	41-60	0.5	2.5	3.0	09:57	12:03	Actual TW (Minutes)	
		0.6	0.0	0.0	10:00	12:00	600	720
Strict	61-80	0.7	-2.5	-3.0	10:03	11:57		
		0.8	-5.0	-6.0	10:06	11:54		
Very Strict	81-100	0.9	-7.5	-9.0	10:09	11:51	*timeDiff*	
		1	-10.0	-12.0	10:12	11:48	120	

For strictness score (*S*) calculation in the function, for instance, the average punctuality score, sales volume, quantity etc. are considered. In the example in **Table 1**, the most flexible user is considered to have an ***s*** value of 0, the offset is 15% of the timeDiff, for which *f* becomes 18 minutes. In the contrast, for most strict instances the *f* is -12 minutes.

For the ***Strict*** and ***Very Strict*** levelled customers the time window becomes even shorter while for the ***Flexible*** and ***Very Flexible*** levelled customers, the time window becomes larger. For ***Normal*** levelled customers, the time window becomes slightly larger or remains unchanged. A correlation between customer satisfaction levels applying **LA** on a soft-time window is shown in **Figure 8**. The red lines denote as hard time windows become shorter.

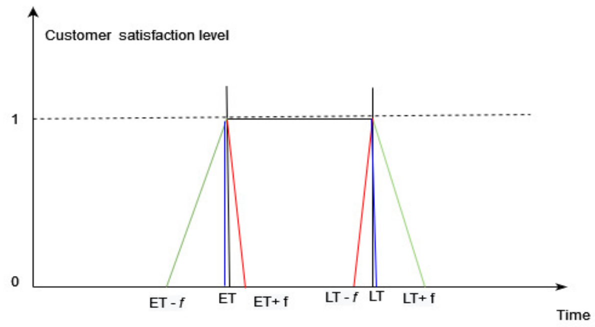

Figure 8: Customer satisfaction level on Soft- time window (applying LA). *The area between ET and LT is the actual (hard) time window. The flexibility offset is ±f. For flexible and very flexible customers the new time window becomes larger, which is shown as an expanded area between **ET-f** and **LT+f**. The blue lines denote the normal levelled customers.*

Let's consider a given time window of a customer, 10:00-12:00, where ET would be then 10:00 and LT=12:00. Depending on the strictness score s (which is derived from several input parameters such as average punctuality, sales volume, mean orders in the past etc.). Let's assume the average punctuality score, S1 is 0.4. Score S2 due to sales volume per quantity from the past is 0.9 and the score from mean order value the S3 lies at 0.8. If the weight factors for the score values are 1 then **Equation 3** can be used to determine the score value as follows:

$$s = (S1.f1 + S2.f2 + S3.f3) / (f1+f2+f3) = (S1+S2+S3)/3$$

The overall strictness score (mean value) for the order is then ***s*** **=** 0.7, which means that the user can be set as a ***Very Strict*** levelled user, the *f* value would be then, for example, -2 minutes. Consequently, the time window can be updated as 9:58-11:58. However, the input parameters as well as the offset setting depend on the implementation perspectives and available constraints. Thus, the output of the algorithm does not guarantee a constant result for the same time window with different dependent variables.

5. Case Study

In this Chapter, the procedures for data collection, data pre-processing, and evaluation, as well as the case study methodology, are outlined. Various cases, their characteristics, and the necessary algorithms to comprehend the application of tools and methods are described. The results of the study are presented graphically and their relevance to the previously formulated research questions is discussed.

5.1 Experimental settings

5.1.1 Data collection and pre-processing

Using MySQL queries several custom views are generated which include customs includes and excludes of datasets. From the core view results in a new Case-study database are generated to hold case-study source evidence.

Table 2: Pseudo query example of data collection.

/** 1. order perspective **/ SELECT order FROM deloma_database WHERE x	/** 2. user perspective **/ SELECT aggregated_user_data FROM deloma_database WHERE y GROUP BY user

Two different query examples are shown as pseudo texts in **Table 2**. Here **x** in the first query stands for custom restrictions applied to collect delivery or order information includes for example: excluding the existing tours which cover more than one day, have missing location data, occurred before 2021 etc. (see more details in Annex, page 47). Whereas the 2nd query in **Table 2** queries the user-centred aggregation data such as average punctual orders, total orders, average ordered items etc. Here y stands also for restrictions such as excluding users who didn’t order, who don’t have the same orders as in queried order dataset, whose location data is missing etc. From these aggregated data instances, it is easy to decide whether the customer is a frequent customer or not.

5.1.2 Data scheme overview

Although several restrictions are applied, while collecting datasets, there are still missing values or unwanted values that go beyond the scope of this case study research. Those data are cleaned and prepared to use as input for the research. For each single queried order instance including the customer's delivery information, all necessary data are checked and then directly saved for evaluation. The class diagram in **Figure 9** shows a partial overview of both the processed input data and output data with relevant attributes. For each case, different results are mapped to the evaluation entity.

The central class in the data scheme is the '*Tour*' class. In general, a tour consists of several stops where the vehicle needs to visit. These stops are defined as an attribute '*stopData*' which is a list of '*TourStopData*' instances. This '*TourStopData*' keeps track of the actual info and the updated info for every stop as well as the order associated with the respective stop of a '*Tour*' instance. The 'Order' class is responsible for the collected dataset consisting of numerical values such as location coordinates, order sales volume

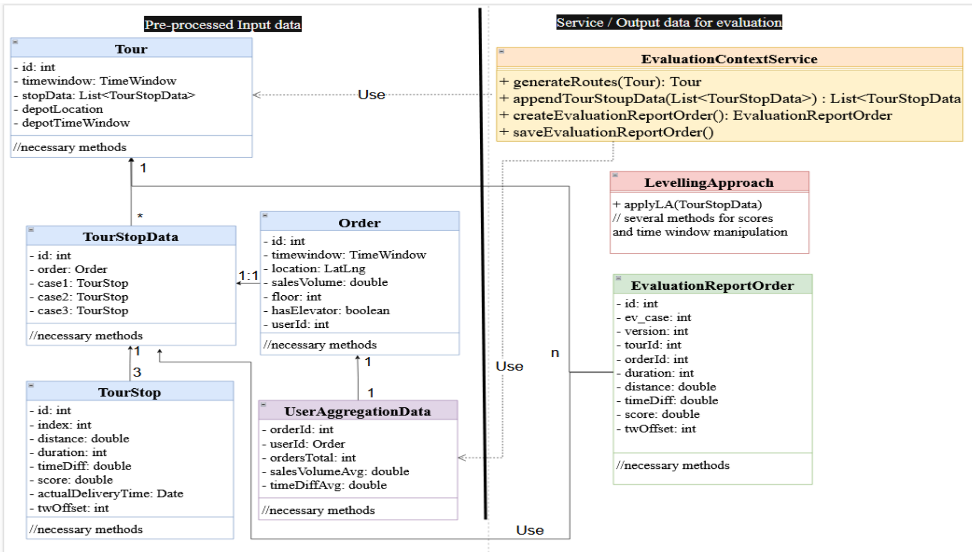

Figure 9: An overview of the data scheme in a class diagram. *To explain how the input and output data look like, only some important classes are shown in the class diagram.*

(e.g., 100€), time window (e.g., 2021-10-29 10:00, 2021-10-29 12:00), customer-specific information such as: whether the customer is new or not, what the provided floor number is as well as the presence of an elevator. Besides, for each stop, the routing information e.g., travel distance, duration, actual delivery time etc. attributes are part of '*TourStop*'.Every 'TourStopData' consists of 3 different versions of 'TourStop' instances, where the first instance is directly derived from the existing dataset, and the second and the third instances are responsible for experimental cases which are explained in the next Sections (see Section 5.2). To use the proposed new Approach (LA), user aggregated data are needed for every current 'Order' instance that holds previous historical information such as total orders, sales volume, and punctuality in the past. The '*UserAggregationData'* class associated these historical data with respective attributes for each ‘Order’ instance. The service class ‘*EvaluationContextService’*, with several methods, is responsible for interacting with all services as well as saving the results of the experiments. In addition to these, for existing tour data as well as for experimental cases, ‘*EvaluationReportOrder’* instances are created and saved in the case study database. These instances include the respective tour’s information (‘tourId’), order’s information (‘orderId’), information for experimental three cases (‘*ev_case’*), their versions as well as some other routing information from the ‘*TourStopData*’ instance. A single ‘Tour’ instance is used several times for the evaluation of different cases (and versions).

However, for evaluation purposes, several versions of attributes are saved in the case study database. These attributes are used to perform a comparative analysis among 3 cases (see Section 5.3). The important attributes are described in the following (see **Table 3**).

Table 3: Important attributes and their descriptions

	Variable / Abbr.	Description
Both	***duration***	The travel duration from the previous location (for the first order depot is the previous location) to the current order location
	distance	Travel distance from the previous location to the current order location.
	timeDiff	The difference between actual delivery time and the provided customer’s hard time window
Input	**timeDiffAvg = t'_{avg}**	Numeric value, in minutes and represents the average of differences between actual delivery and provided hard time window, derived from the historical data for the customer of placed order.
	N_o	Number of total orders
	orderTotal **=** N_{o_hist}	Numeric value, number of total orders in the past order history.
	salesVolumeCurrent = $S_{vol\text{-}current}$	Numeric value, sales volume value of the current order.
	salesVolumeAvg = $S_{vol\text{-}hist\text{-}avg}$	Numeric value, avg sales volume value from the historical data for the customer.
	elevator = *E’*	Denotes whether the customer has an elevator or not.
	floor* = *F’	The numeric value and floor number are given to the customer.
	customerType **= Ctype**	Denotes whether the customer is a commercial or private individual.
Output	***score***	Strictness score for the order/user in this current order
	twOffset	Offset in minutes that are to be added or deducted while manipulating the time window
	N_{po}	Number of punctual orders per tour
	R_{po}	The ratio of punctual orders per tour
	S_{dist}	Sum of travel distance per tour
	S_{dur}	Sum of travel duration per tour
	R_{po_mean}	Mean punctual orders
	S_{dist_mean}	Mean of overall travel distance (Km)
	S_{dur_mean}	Mean of overall travel duration (minutes)

5.1.3 Expectations of the evaluation

The mean value of input data is R_{po_mean} = 8.91, S_{dist_mean} = 104.29 and S_{dur_mean}= 162.71. The evaluation in this study focuses based on the research question by comparing the generated test results against these mean values. For evaluation, the research questions are reformulated with variables and stated in the following.

Reformulated Research questions:

a) Do the mean values of distance (S_{dist_mean}) and duration (S_{dur_mean}) decrease?

b) Does the mean punctual orders (R_{po_mean}) increase?

1. What is the effect of using the Jsprit tool?
2. What is the effect of applying LA based-on input parameters, $S_{vol-current}$, $S_{vol-hist-avg}$, N_o, ***Ctype*** and ***E'*** and ***F'***?

In this study, for the summary of the evaluation, all mean values (S_{dur_mean}, S_{dist_mean} and the R_{po_mean}) are to be compared for both cases. More details about the evaluation cases are described in the following Sections.

5.2 Cases

Following the research design, this study compares and assesses three distinct cases: **Case 1**: *Manual route planning* (currently existing version), **Case 2**: *Integrating Jsprit*, and **Case 3**: *Applying LA*. All these cases have distinct outcomes that require comparison and evaluation. Below is a brief explanation of all three cases.

5.2.1 Case 1: *Manual route planning* (The currently existing version)

This case refers to the case where the routing information exists already which means the routing sequence, duration, and distance of routes as well as the time differences among actual delivery times and the preferred time window are already there. Calculating the individual distance, duration as well as time differences among routed locations, overall information is saved into an evaluation report. Two other cases are compared to this one for evaluation and then comparative results are described in the next Sections.

5.2.2 Case 2: *Integrating Jsprit*

Jsprit already includes robust solution strategies. In Jsprit, it is necessary to set vehicles and services to generate routes for a given problem set and known constraints. The default number of vehicles in a fleet is infinite, indicating that multiple vehicles will be available. This setting, however, exceeds the constraints of this study. Consequently, the fleet size has been fixed. Jsprit algorithm strategy finds the best of all possible solutions which contain a list of assigned or unassigned jobs (see Algorithm 4). Assigned jobs are the locations to be served and set into a routing sequence while unassigned jobs are dropped locations, that are not possible to fit into the route.

Algorithm 2. **GENERATE ROUTES**

Input: *List of Tour*

Output*: List of Tour*

1. *Initialization of variables: initialize variables routes*
2. ***for each*** *Tour (**t**) // iterate the input list*
3. ***routes*** ← *findRoutingSolutions (tour, withOutLA)*

   //Check the algorithm attached.
4. *appendUpdatedRouteData(routes, withOutLA)*
5. *End*

As the vehicle starts from the depot, the start time, and location is also set for the vehicle. The constraint also added that the vehicle must not return to the depot. A constant capacity is set for the vehicle. Then for all delivery locations, services are created. Algorithm 5 explains the creation of services in which each delivery location, time window, fixed demand and service time are added. The found solution gives a list of locations in an ordered sequence with actual delivery time and the time-of-service end. With the help of the Distance Matrix API call, distance and duration among places are calculated. All this information is necessary to create evaluation reports. For all collected data instances, the tours are generated. In this case, for each tour, the generated routing information is saved (see Algorithm 2).

5.2.3 Case 3: Applying LA

The third case is the extension of *Integrating Jsprit* (Case 2) where LA (described in Section 4.4) is applied. This Case is divided into 3 versions (V1, V2 and V3). Considering the input parameters, 5 score functions, their weight factor and constants are defined. In **Table 4** In the first version (V1), only punctuality in the historical data is considered. Therefore, the score function: *scorePunctuality* (S_p) is to be calculated. Only the weight factor for punctuality (f_p) is set to 1, and the rest of the factors remain at 0. After that, for version 2 (V2), sales volumes ($S_{vol\text{-}current}$ and $S_{vol\text{-}hist\text{-}avg}$) and the number of historical orders (N_{o_hist}) are considered. Hence, only S_{OT} and S_{SV} are set to 1 and the rest of the factors remain 0. Finally, for V3, all factors are set to 1, except f_{EF}.is set to 0.5. The constants stated in **Table 4** define the limit at which or above, the scores are to be at 100%. Below these values the score can be then 0 – 99.9%, e.g., the score can be 0-99.9% only if the average punctuality (t'_{avg}) is less than N_{maxT} =30 minutes. These constants are used in the upcoming Sections to calculate scores.

Table 4: Score function, weight factor and constants

Score function (Abbr.)	***Weight factor***	**Constants**	**Description**
scorePunctuality (S_p)	f_p	*Maximum time difference* (N_{maxT}) = **30** (minutes)	Score and factor for average punctuality in the historical data.
scoreSalesVolume (S_{sv})	f_{sv}	*Maximum sales volume* (S_{vol_max}) **= 175 €**	Score and factor in current and average past sales volume values.
scoreOrderTotal (S_{OT})	f_{OT}	*Maximum number of orders* (N_{maxo}) **= 50**	Score and factor for the number of orders in the past for the customer
scoreCustomerType (S_{CT})	f_{CT}	-	Score and factor depending on customer type
scoreElevFloor (S_{EF})	f_{EF}	*Maximum number of floors* **(maxFlr) = 4**	Score and factor for floor number and presence of an elevator

5.2.3.1 How *ScorePunctuality (Sp)* is to be calculated

Based on the input parameter (number of average punctualities in the past order history) of the specific user, the score value is defined using the following linear equation:

Equation 6: Calculating *scorePunctuality* (S_p)

$$S_p(t'_{avg}) = \frac{1}{N_{\mathrm{maxT}}}|t'_{avg}|$$

5.2.3.2 How S_{sv} is to be calculated

Based on the input parameters $S_{vol\text{-}hist\text{-}avg}$, $S_{vol\text{-}current}$ and the score value is defined using the following linear equations:

Equation 7: Calculating *scoreSalesVolumeCurrent* ($S_{sv\_current}$)

$$S_{sv\_current} = \frac{1}{S_{vol\_max}} S_{vol-current}$$

Equation 8: Calculating *scoreSalesVolumeHistAvg* ($S_{sv\_hist\_avg}$)

$$S_{sv\_hist\_avg} = \frac{1}{S_{vol\_max}} S_{vol-hist-avg}$$

Equation 9: Calculating *scoreSalesVolume* (S_{sv})

$$S_{SV}\left(S_{sv\_current}, N_{o_{hist}}, S_{sv\_hist\_avg}\right) = \frac{(1 * S_{sv\_current} + N_{o\_hist} * S_{sv\_hist\_avg})}{(N_{o\_hist} + 1)}$$

Here, $S_{sv\_current}$ results in a score for the sales value of the current order, while $S_{sv\_hist\_avg}$ is for the historical data. Finally, the total score for sales volumes is calculated.

5.2.3.3 How S_{OT} is to be calculated

Based on the input parameter N_{o_hist} of the specific user, the score value is defined. Let's say the user has no orders in the past at all, which means the current order is the first order and the user is new in this case. The author decided to hold the strictness score for this new User as 100% strict. On the contrary, for users having several orders which are more than or equal to one order, the score is calculated using the following linear equation:

Equation 10: Calculating *scoreOrderTotal* (S_{OT})

$$\boldsymbol{S_{OT}(N_{o\_hist}) = \frac{1}{N_{\mathbf{maxo}}} N_{o\_hist}\ ;\ N_{o\_hist} \geq 1}$$

For new customers, only if $N_{o\_hist} = 0$ then $\boldsymbol{S_{OT}}$ is set to 1.

5.2.3.4 How the S_{CT} is to be calculated

Based on the KPI-defining opinion of the company's representative (see page 47 for details), the $\boldsymbol{S_{CT}}$ value is set to -0.5 for commercial and 1.0 for private customers.

5.2.3.5 How S_{EF} is to be calculated

Based on the input parameter (number of average punctualities in the past order history) of the specific user, the score value is defined using the following equation:

Equation 11: Calculating score for elevator and floor, *scoreElevFloor* (S_{EF})

$$\boldsymbol{S_{EF}}(E', F') = \begin{cases} 0.75\ if\ E' = 1 \\ 1 - \dfrac{1}{maxF\boldsymbol{lr}} F';\ and\ 0 \leq \left|\dfrac{F'}{\boldsymbol{maxFlr}}\right| \leq 1 \end{cases}$$

5.2.3.6 Finding the Final score and flexibility constraint *f*

The weighted total score can be derived by using the following equation:

Equation 12: Calculating weighted total score (S_f)

$$S_f = S_p S_p + S_{sv} f_{sv} + S_{OT} f_{OT} + S_{CT} f_{CT} + S_{EF} f_{EF}$$

The final score can be obtained by using **Equation 2** and **Equation 3**. During the LA application, 3 different versions are tested the coefficient values are chosen according to the questions in the previous Section. Based on the final score, using **Equation 4** and **Equation 5**, the flexibility offset is determined, and the respective time windows are updated. Findings due to the experimental parameter settings are compared and discussed in the next Sections. The slope for the **Equation 4** was selected -25 and b = 15.

5.3 Results

This Section presents the study's findings. This component of the results is spitted into two sub-Sections: a comparative analysis of different cases and a discussion. The first Subsection describes the results of the case study including visual representations of the comparative results of all cases, highlighting the key differences as well as a summary of overall results. Finally, the key learnings as well as a brief discussion of the results are stated in the second Subsection. Please refer to Annex (pages 48-52) for detailed graphical and tabular representations of the results.

5.3.1 *Manual route planning (Case 1)* Vs *Integrating Jsprit (Case 2)*

Here the result of *Manual route planning* (Case 1) and *Integrating Jsprit* (Case 2) are compared. In both cases, routing solutions are generated by considering customer provided hard time window. In this Section, initially, 16 tours are visualised to show how the duration (**S_{dur}**), distance (**S_{dist}**) and the number of punctual orders (**N_{po}**) are changed in *Integrating Jsprit* (Case 2). Finally, an overview of the results is given. For most of the tours, the number of punctual orders (N_{po}) becomes higher than in *Manual route planning* (Case 1) by integrating Jsprit. The orange line in **Figure**

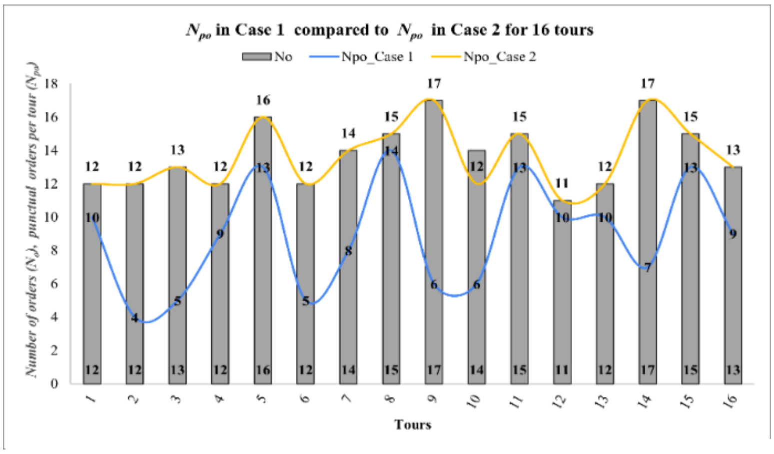

Figure 10: Comparison of the number of punctual deliveries per tour in Manual route planning vs Integrating Jsprit (Case 2). *The total number of orders per tour No is plotted as columns. The blue line refers to punctual orders in Case 1 and the orange line refers punctual orders in Case 2.*

10 indicates that for some tours, even the value becomes 2-3 times higher This indicates an increase in the number of punctual orders (N_{po}) increase when *Integrating Jsprit* (Case 2). In both **Figure 11** and **Figure 12** graphics, the total travel distance (S_{dist}) and duration (S_{dur}) per tour remain almost similar ranges except for some tours, a slight increment in

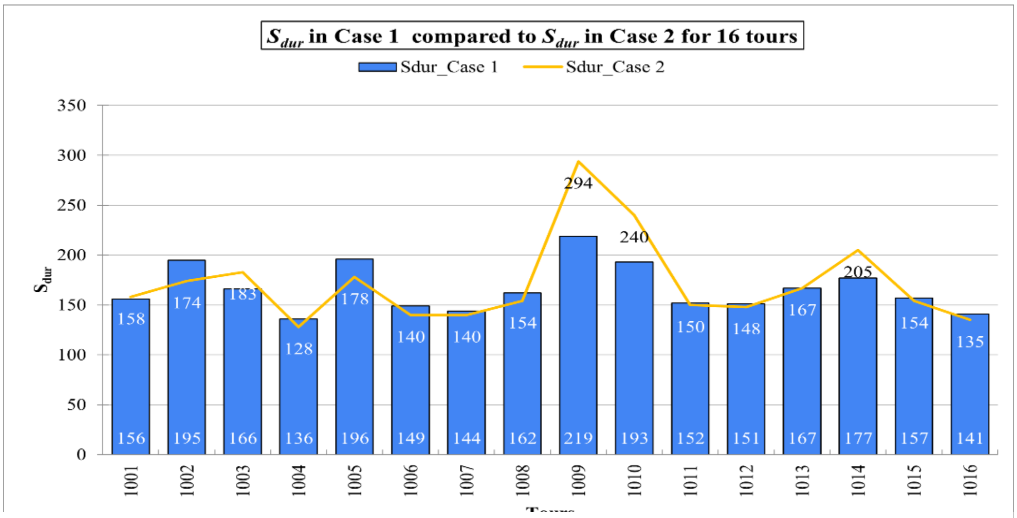

Figure 11:Travel duration comparison of all cases (16 tours). *The graph shows a comparison of the total travel distance per tour. Here, the sum of duration Sdur in minutes is plotted on the vertical axis. Case 1 is represented by the blue columns and Case 2 by the orange line.*

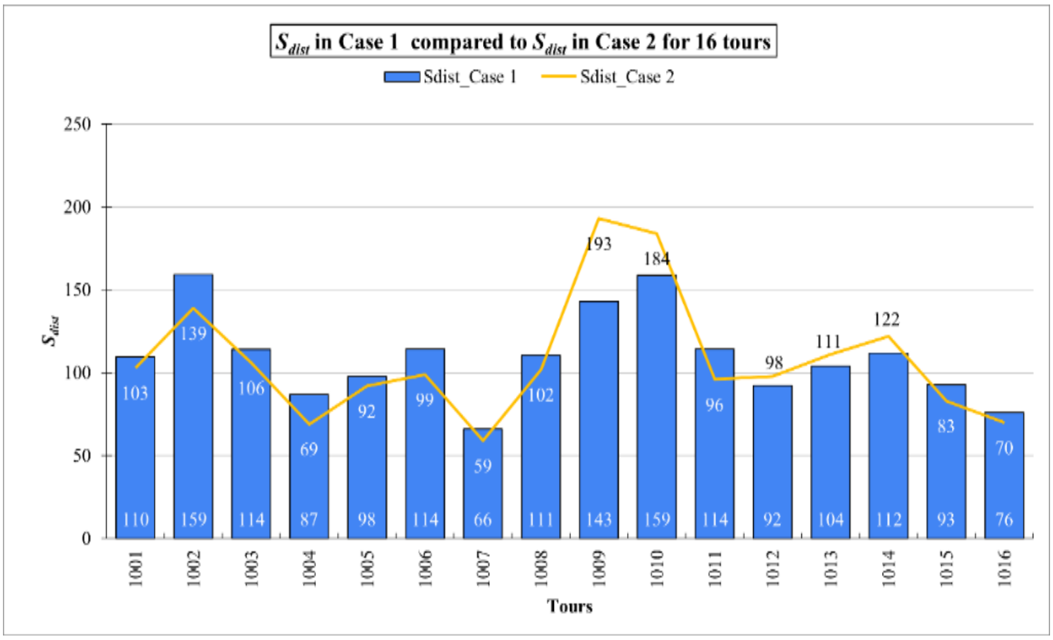

Figure 12: Travel distance comparison in Case 1 vs Case 2 (for 16 tours). *The graph shows a comparison of the total travel distance per tour for both cases. S_{dist} in kilometres (km) is plotted on the vertical axis. Case 1 represents the columns coloured blue and orange line for Case 2. On the horizontal axis, respective tour ids are shown.*

travel duration is seen when Jsprit is integrated. The values increase in 5 out of 16 tours compared to *Manual route planning* (Case 1).

- **Overall overview:**

For **400** tours and **5602** order instances, *Integrating Jsprit (Case 2)* shows about 1996 more punctual orders compared to *Manual route planning* (Case 1). As a result, the mean punctual order (R_{po_mean}) increases by about 56% compared to *Manual route planning*. Although the mean travel duration (S_{dur_mean}) increases by about 1%, the mean travel distance (S_{dist_mean}) decreases by about 4%.

5.3.2 *Manual route planning (Case 1)* Vs *Applying LA (Case 3)*

The previous Section indicates that *Integrating Jsprit* (Case 2) does better than Manual route planning (Case 1) in terms of punctual deliveries and travel distance reduction, but the mean travel duration increases slightly.

In this case, instead of focusing just on overall improvements, LA defines each order as a specific score-based time window manipulation. This means instead of focusing just on overall punctuality, travel distance and duration, the algorithm decides to penalize a low-scored customer and prioritizes high-scored customers. As a result, in contrast to 'Integrating Jsprit' (Case 2), the evaluation expects to be changed internally due to soft time windows. For better understanding, the levels represent the score ranges and customer priority. In *Applying LA (Case 3),* 3 different versions are experimented with, and their results are compared to *Manual route Planning* and described in the following.

5.3.2.1 Case 1 vs Case 3V1

In this specific version only the input attribute is t'_{avg} set to be active, and the rest of the attributes are ignored. Based on the scores, the orders are prioritized. As the maximum time offset for strictness N_{maxT} for ***ScorePunctuality (Sp)*** in this study is set to 30 minutes (see in Section 0), the orders having an absolute value of greater than and equal to 30 minutes are to be *VERY STRICT* levelled. On the contrary, the customers who have a good or very good punctuality score (e.g., up to 12 minutes) are to be penalized such that other customers (orders) get more punctual deliveries. In

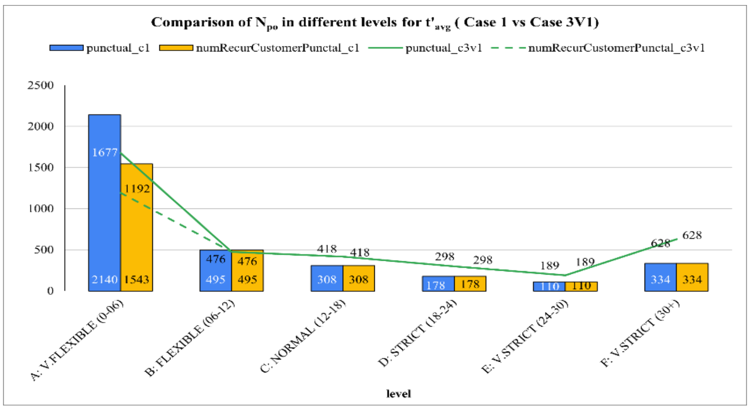

Figure 13:Punctual order distribution based on levels (Case 1 vs Case 3V1 comparison). *The graph shows a comparison of N_{po} between Case 1 and Case 3V1. The columns represent the number of total orders out of which the punctual orders are denoted using lines.*

Figure 13, the green line indicates that the punctuality drops for low-scored orders (*FLEXIBLE* and *VERY FLEXIBLE* levelled orders). Therefore, the orders having average or higher scores (*NORMAL*, *STRICT* and *VERY STRICT* levelled orders) are more punctual

5.3.2.2 Case 1 vs Case 3V2

In this version, only *scoreSalesVolume* (S_{sv}) and *scoreOrderTotal* (S_{OT}) are considered. Based on the constants, maxo and S_{vol_max} values (see 5.2.3.2 and 5.2.3.3), the orders (users) are levelled for specific ranges. The visual graphic shown in **Figure 18** explains how the *scoreOrderTotal* (S_{OT}) influences the overall results. The results in **Figure 17** and **Figure 19** indicate how the number of punctual orders (N_{po}) increases or decreases in *Applying LA (Case 3 V2)*. Because of the score *scoreOrderTotal* (S_{OT}), new customers get prioritized, on the contrary for sales volume scores recurring customers get prioritized as well. Lower or below-average orders with below-average sales volume get penalized, therefore the orders with higher sales volume become more punctual.

5.3.2.3 Case 1 vs Case 3 V3

In this version, all 5 scores and factors (which were introduced in Section 5.2.3) are considered. As individual score values have a direct influence on the overall score, the results show significant differences among the value of output attributes. Levelling based on historical time difference (t'_{avg}) ranges in **Figure 14** the changes in punctual order ratio are remarkable. Not only did about 63% of *VERY STRICT* levelled orders become more punctual but also the *FLEXIBLE* and *VERY FLEXIBLE* levelled orders show significant ratio op punctual order (*Rpo*) changes. The sales volume scores are the key changer in overall scores, in **Figure 21** and **Figure 22** shows how the sales volume value affects the overall punctual order ratio (R_{po}). Not only the recurring customers but also the new customers with higher sales volume get prioritized. However, the sales volume scores are dependent on the number of orders

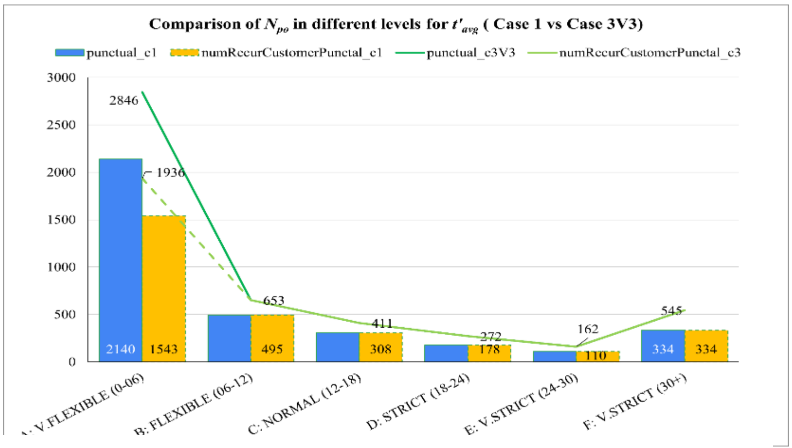

Figure 14:Comparison of Npo in different levels for t'_{avg} (Case 1 vs Case 3V3)

(N_{o_hist}) too. In **Figure 20**, it is shown that for the new customers (customers' first orders), the *Rpo* value rises by about 52%. For recurring customers, for example, who ordered 1 to 10 orders in the past, the *Rpo* value rises to its peak value of almost 32.6%. Both for commercial customers and private customers, the ratio of punctual orders (*Rpo*) increases to 16.3% and 45% (see **Figure 23**). How S_{EF} affects the whole score, is shown in **Figure 24**. For *NORMAL* and *STRICT* levelled orders, R_{po} increases by about 32- 34 %. For *VERY STRICT* levelled orders rises to 44%.

5.3.3 Overall results summary

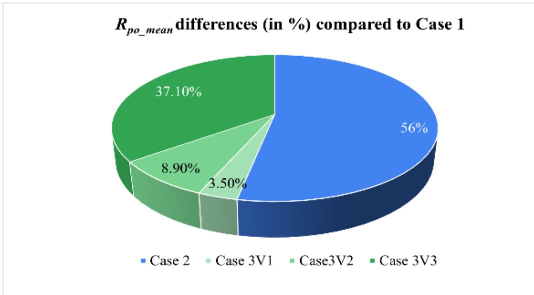

Figure 16: R_{po_mean} differences compared to Case 1.

In *Integrating Jsprit* (Case 2), as the hard time windows are set. The routing algorithm doesn't consider other customer-centred factors (for example the priority of sales volume, the importance of early arrival etc.). Hence, the mean of punctual orders (R_{po_mean}) increases by almost 56.0% more than in *Manual route planning* (Case 1) (see **Figure 16**). In *Applying LA (Case 3 V3)*, the mean punctual orders (R_{po_mean}) reach the second highest position holding at 37.1%. The S_{dist_mean} value decreases in Case 2 as well as in all *Applying LA (Case 3)* versions (see). The most decremented value of mean travel distance (S_{dist_mean}) by about 10.3%, is seen in both *Applying LA (Case 3 V2)* and *Applying LA (Case 3 V3)*. The S_{dur_mean} value decreases in *Applying LA (Case 3)*, but only in *Integrating Jsprit* (Case 2), it increases by about 1.0%. The most decremented value of S_{dur_mean} is found in *Applying LA (Case 3 V2)*. However, the overall results in *Applying LA (Case 3)* are affected due to internally score-based R_{po} improvement. By applying LA, depending on chosen several factors, the versions in *Applying LA (Case 3)* show remarkable differences.

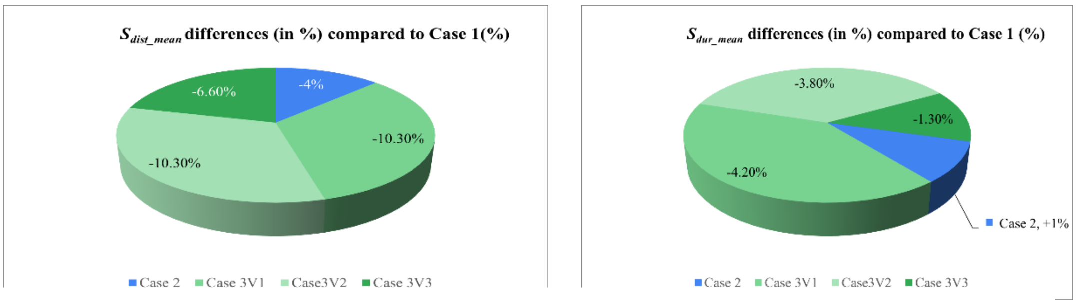

Figure 15: Overall mean distance (S_{dist_mean}) and duration (S_{dur_mean}) differences compared to Case 1. *The left graphic shows the differences for S_{dist_mean} and the right graph for S_{dur_mean}.*

5.4 Discussion

This Section briefly analyses these results and their conclusions to investigate the effectiveness of the used tool and the algorithms in this study. To investigate the improvement of the service, some research questions were set and stated in Section 3.1. For in-depth analysis, the reformulated research questions introducing input and output variables for evaluation were addressed in Section 5.1.2. For comparing the effectiveness of the tool and LA, about 400 tours were regenerated. After that, the existing data were compared with newly generated data. In the previous Sections and Annex, the settings of the case study and test results were addressed. To test the null hypothesis (see Section 3.2), several t-tests were conducted for paired two independent samples. Each pair consisted of data from *Manual route planning* (Case 1) as the first sample, while the second sample was then chosen as data from *Integrating Jsprit* (Case 2) and *Applying LA (Case 3)* (all versions).

The case study of vehicle route planning incorporating Jsprit for Deloma UG revealed several key findings. First, *Integrating Jsprit* alone has already produced better results compared to *Manual route planning*. Second, when *applying LA*, the results got significant improvement in terms of mean duration and distance. With soft time windows, user-centred time window flexibility has shown significant influences, e.g, in Case 3V3, because of LA, internally the customers have higher order value and recuring gets more punctual order (see pages 48-52).

The results of the t-tests (see **Table 10**) showed that there were statistically significant differences in mean punctual orders (R_{po_mean}), mean travel duration (S_{dur_mean}) and mean travel distance (S_{dist_mean}) among paired samples with a degree of freedom, $df = 399$ and significant value, p = 0.05. Specifically, the mean travel distance (S_{dist_mean}) was significantly lower in Case 2 and all of *Applying LA (Case 3)* versions than the existing mean distance value of *Manual route planning* (Case 1). On the other hand, for mean travel duration (S_{dur_mean}), the differences are statistically significant only for *Applying LA (Case 3)* (for all versions V1, V2 and V3). In *Integrating Jsprit* (Case 2), instead of decreasing, the mean travel duration (S_{dur_mean}) increases slightly. The mean punctual orders (R_{po_mean}) for *Integrating Jsprit* (Case 2), *Applying LA (Case 3 V2)* and *Applying LA (Case 3 V3)* were significantly higher than the existing mean punctual orders (R_{po_mean}) of *Manual route planning* (Case 1). But for *Applying LA (Case 3 V1)*, the R_{po_mean} values increased insignificantly. The null hypothesis H_0 restrained only for S_{dur_mean} in

Integrating Jsprit (Case 2) and for R_{po_mean} in *Applying LA (Case 3 V1)*. For *Applying LA (Case 3 V2)* and *Applying LA (Case 3 V2)*, H_0 can be rejected. Consequently, the alternative hypothesis H1 remains to be restrained.

In the first research question (see Section 5.1.3 **a**), the decrement in total travel duration and distance was concerned. The mean travel distance (S_{dist_mean}) decreases for all cases and versions which were discussed with a comparative analysis in Section 5.3.1. Incorporating Jsprit not only automated route planning but also the S_{dist_mean} decreased significantly for example for *Integrating Jsprit* (Case 2) by up to 4%. From the comparisons in Section 5.3.2 and Table 11, the data indicates that for all 3 versions in *Applying LA (Case 3)*, the S_{dist_mean} decreased as well. From the same table and Section, the results show a rise of S_{dur_mean} by about 1.11% for *Integrating Jsprit* (Case 2), while comparing with *Applying LA (Case 3)* (all versions), the value decreases up to 4.2%. By reducing the travel distance and travel duration, *Applying LA (Case 3)* shows a better result and hence the study fulfils the first objective to answer the first research question.

The second research question (see Section 5.1.3) raised in this study is concerning customer satisfaction improvement by increasing the ratio of punctual order deliveries. On the other hand, due to the input parameter t'_{avg} consideration, to answer the question stated in Section 5.3.2. As the result, internally almost 22% of the lowest scored (*VERY FLEXIBLE* levelled are penalized, therefore almost 88% of *VERY STRICT* levelled orders get more punctual. But in *Applying LA (Case 3 V3)*, due to other score values, the *Rpo* increases internally for all levelled orders. As a result, the overall result indicates a rise of about 23.3% for the R_{po_mean} value. Compared to *Manual route planning* (Case 1), the result data of *Integrating Jsprit* (Case 2) and both *Applying LA (Case 3)* versions show a higher value of R_{po_mean} (see Section 5.3.3). Hence, the second research question is answered. The results of this study are partially dependent on how which attributes are to be defined for Key Performance Indicators (KPI). For instance, by choosing maxT = 30, the customers having more than 30 minutes of average delays in the historical data, get more prioritized. So, for the same data instances, with different factors (see **Table 9**), the results can be different. To use LA, from this study's key learning, it's recommended to check important factors and score values concerning the company's KPI definition.

6. Limitations

The incorporation of Jsprit resulted in an improvement in punctual orders. The proposed LA was able to impact positively internally customer's prioritizing which could lead to enhanced customer satisfaction. However, some limitations are found which should be considered when interpreting the findings of this study. To begin with, the data for this study were collected under a Non-Disclosure Agreement from a German corporation named Deloma UG. Due to confidentiality protection, the results cannot be generalized to other corporations or made public. More specifically, the data from only 3 local online shops were used. Hence, a generalized evaluation was not possible.

Second, about 400 tours (5602 orders) were considered for the test sample. These data consist of only of 16 months (January 2021 - June 2022). Considering this sample, analysis of perspectives for the LA application (see 5.4 for details), may not be representative of the larger dataset. To apply LA, historical data played a vital role in score functions. Hence, further research with a larger dataset would be necessary to confirm and expand upon these findings. Besides, this study was limited to a three-month time frame. This may not provide a complete picture of the approach under investigation and may not be representative of longer-term studies.

Third, access to the Google Distance Matrix API is limited to requests (maximum 100 elements per request). Although per month 200$ Google Maps Platform credit is included for the free tier, this may not be feasible or enough for all researchers to pay the extra costs for further usage. Consequently, this may limit the accessibility of the data to those with the resources to pay for it. The travel distance and duration were calculated using this API, which is assumed to be accurate and up-to-date.

Finally, this study was limited to the single depot, single vehicle and single time window provided by the users. The accuracy of vehicle load time, unload time, and service durations were considered under certain assumptions (see Section 4.4 and 5.1.1). The statistical tests (t-tests) used in the analysis made assumptions about the data (the independence of the samples). The result could be deceptive if these assumptions were not met. Moreover, the choice of VRP solving tool was limited to Jsprit. To measure the effectiveness of LA, further research with a longer temporal scope may be necessary to confirm or refute the findings of this study.

7. Conclusion and Future Work

In this case study, the effectiveness of integrating the Jsprit tool and the proposed approach LA is evaluated. By comparing existing manual tour data with regenerated routing data, overall punctuality was improved. Considering the importance of customer satisfaction as well as the defined KPIs, LA can internally offer flexible time windows. Considering user-centred input parameters the LA generates a score for orders and manipulates time windows. Thus, the number of punctual orders increases and at the same time distance and duration decrease. The results of this study suggest that a well-balanced scoring depends on KPI definition as well as the user’s historical data and based on that LA can be applied to enhance delivery service quality. Overall, this study contributes to a better understanding of user-centred soft time window manipulation concerning LA.

There are several possible future research areas to mention from the learning and shortcomings of this study. For example, incorporating other alternative route solver tools along with LA and introducing user rating score functions, instead of linear functions, exponential functions in score calculation etc. A case study with a larger data set, including more historical data and working with multiple companies’ KPIs could be an interesting extension of this study. Additionally, it would be valuable to research other VRP variants. The possible factors of satisfaction introduced at the very beginning of this paper can be considered in the extension of LA. These factors would then possibly impact business growth.

Despite the study's limitations, it might be claimed that the outcome of this study was effective, and the targeted goal was met. The applicability of these findings to other organizations and industries requires additional research. However, the success of LA on a specific dataset in Deloma UG suggests that this solution has the potential to bring similar benefits to other organizations facing similar challenges.

References

Alexander, J., Markus, F. & Rainer, K., 2020. *The vehicle routing problem with time windows, flexible service locations and time-dependent location capacity,* Munich: Technical University of Munich,.

Anon., 2020. *Was Online-Kunden wünschen.* [Online]
Available at: https://www.handelsjournal.de/dossiers/corona/2020/september/was-online-kunden-wuenschen.html
[Accessed 26 9 2022].

Ayman, M., Tarek, C., Sebastian, H. & Jakob, P., 2022. *Extending JSprit to solve electric vehicle routing problems with recharging.* Porto, Elsevier B.V.

Charlotte, K., Jan, F. ,. & Ann, M. C., 2020. Flexible time window management for attended home deliveries. *omega,* 03.Volume 91.

Dantzig, J. R., 1959. THE TRUCK DISPATCHING PROBLEM.. p. 80–91.

Desaulniers, G., Madsen, O. B. & Ropke, S., 2014. The Vehicle Routing Problem with Time Windows. In: T. Paolo & V. Daniele, eds. *Vehicle Routing Problems, Methods, and Applications.* Bologna: SIAM, pp. 119-159.

Éric, T. et al., 1997. A Tabu Search Heuristic for the Vehicle Routing Problem with Soft Time Windows. *Transportation Science,* 31(2), pp. 170-186.

Eurostat, 2022. *Impact of COVID-19 on e-sales of enterprises.* [Online]
Available at: https://ec.europa.eu/eurostat/statistics-explained/index.php?title=Impact_of_COVID-19_on_e-sales_of_enterprises
[Accessed 10 11 2022].

Eurostat, 2022. *SMEs showed resilience to effects of pandemic.* [Online]
Available at: https://ec.europa.eu/eurostat/en/web/products-eurostat-news/-/ddn-20221028-3 [Accessed 10 11 2022].

Febrian, A. & Fadly, M., 2021. The Impact of Customer Satisfaction with EWOM and Brand Equity on E-Commerce Purchase Intention in Indonesia Moderated by Culture. *Binus Business Review*, 12(1), p. 41–51.

G. Clarke & Wright, J., 1964. Scheduling of Vehicles from a Central Depot to a Number of Delivery Points. Volume 12, no.4, pp. 568-581.

Gilbert, L., Stefan, R. & Thibaut, V., 2014. Heuristics for the Vehicle Routing Problem. In: T. Paolo & V. Daniele, eds. *Vehicle Routing Problems, Methods, and Applications.* Bologna: SIAM, pp. 87-116.

Hokey, M., 1991. A Multiobjective Vehicle Routing Problem with Soft Time Winnows: the Case of a Public Library Distribution System. *Socio-Economic Planning Sciences,* 25(3), pp. 179-188.

IBM Cloud Education, I., 2020. *API.* [Online] Available at: https://www.ibm.com/cloud/learn/api [Accessed 18 11 2022].

J. J., H. & D. W., T., 1985. "Neural" computation of decisions in optimization problems. *Biological Cybernetics volume ,* Volume 52, p. 141–152.

Jacek, W., 2020. Current and emerging formulations and models of real-life rich vehicle routing problems. In: N. Jakub, ed. *Smart Delivery Systems: Solving Complex Vehicle Routing Problems.* Poland: Elsevier , p. 286.

Jan, K. L. & H. G., R. K., 1981. Complexity of vehicle routing and scheduling problems. *Networks,* 11(2), pp. 221 - 227.

Jean-Francois, C., Guy Desaulniers Jacques, D., Marius, M. S. & Francois, S., 2002. VRP with Time Windows . In: T. Paolo & V. Daniele, eds. *THE VEHICLE ROUTING PROBLEM.* s.l.:SIAM, pp. 157-194.

John, H. H., 1992. Genetic Algorithms. *Scientific American,* 267(1), pp. 66-73 .

Karkula, M., Jerzy Duda & Iwona Skalna, 2019. Comparision of capabilities of recent open-source tools for solving capacitated vehicle routing problems with time windows.

Khristianto, W., Suyadi, I. & . K., 2012. The Influence of Information, System and Service on Customer Satisfaction and Loyalty in Online Shopping of Forum Jual Beli Kaskus.Us, Malang Region. *International Journal of Academic Research,* 2(4), pp. 28-32.

Kirkpatrick, S., C. D. Gelatt, J. & Vecchi, M. P., 1983. Optimization by Simulated Annealing. *Science,* 220(4598), pp. 671-680.

Marc, R., Karl, D. & Richard, F. H., 2004. D-Ants: Savings Based Ants divide and conquer the vehicle routing problem. *Computers & Operations Research,* 31(4), pp. 563-591.

Nacima, L., Christian, P. & Caroline, P., 2016. *Metaheuristics for Vehicle Routing Problems.* s.l.:ISTE; John Wiley & Sons, Inc..

Nalepa, J. & Miroslaw, B., 2014. Co-operation in the parallel memetic algorithm, International Journal of Parallel Programming. *International Journal of Parallel Programming volume,* Volume 43, pp. 812-839.

Nebojša, V., Milorad, K. & Kaurin, T., 2019. The Influence of Online Shopping Determinants on Customer Satisfaction in the Serbian Market. *Journal of Theoretical and Applied Electronic Commerce Research,* 14(2), pp. 72-82.

NEO Research Group, U. o. m. s., 2013. *simulated-annealing.* [Online]
Available at: https://neo.lcc.uma.es/vrp/solution-methods/metaheuristics/simulated-annealing/
[Accessed 5 11 2022].

Pappalardo, M., 2021. *Dynamic and stochastic vehicle routing problem: a simulation tool for a large-scale study case,* s.l.: s.n.

Parcel Perform, zenloop, 2020. *insights: E-Commerce Delivery Report: One quarter of German and UK consumers not satisfied with parcel delivery.* [Online]
Available at: https://www.parcelperform.com/insights/e-commerce-delivery-report-german-uk-consumers
[Accessed 10 11 2022].

Patel, R., 2022. *travelling-salesman-problem.* [Online]
Available at: https://www.upperinc.com/guides/travelling-salesman-problem/
[Accessed 13 11 2022].

Patrick, S., 2016. Simulated Annealing. In: S. Patrick, ed. *Metaheuristics.* Switzerland: © Springer International Publishing, pp. 19-50.

Paweł, G. et al., 2020. On a road to optimal fleet routing algorithms: a gentle introduction to the state-of-the-art. In: N. Jacub, ed. *Smart Delivery Systems Solving Complex Vehicle Routing Problems.* s.l.:Elsevier, pp. 37-92.

Rafael, S. V., 2020. *A pragmatic approach to improve the efficiency of the waste management system in Stockholm through the use of Big Data, Heuristics and open source VRP solvers,* Stockholm, Sweden: KTH Royal Institute of Technology.

Ricardo, E., Kai, M.-T., Carina, T. & Kai, N., 2020. *Using a Route-based and Vehicle Type specific Range Constraint for Improving Vehicle Routing Problems with Electric Vehicles.* Paphos, Cyprus, ELSEVIER B.V..

Robert, K. Y., 2016. *Case Study Research and Applications Design and Methods.* Thousand Oaks(California): SAGE (COSMOS Corporation).

Schrimpf, G., Schneider, J., Stamm-Wilbrandt, H. & Dueck, G., 2000. Record Breaking Optimization Results Using the Ruin and Recreate Principle. *Journal of Computational Physics*, 159(2), p. 139–171.

Solomon, M. M., 1987. Algorithms for the Vehicle Routing and Scheduling Problems with Time Window Constraints. *Operations Research*, 35(2), p. 254–265.

Statista, 2022. *COVID-19: e-commerce growth in Europe 2020-2021 | Statista.* [Online]
Available at: https://www.statista.com/statistics/1189076/covid-19-e-commerce-growth-europe-country/
[Accessed 26 09 2022].

Stefan, S., 2016. *Github/Jsprit/SchrimpfFactory.java.* [Online]
Available at: https://github.com/graphhopper/jsprit/blob/master/jsprit-core
[Accessed 19 11 2022].

Tang, Y. & Huang, G., 2018. Vehicle Routing Problem for perishable food in Cold Chain with Fuzzy Time Windows. In: Paris, France: Atlantis Press.

Tilk, C., Olkis, K. & Irnich, S., 2021. The last-mile vehicle routing problem with delivery options. *OR Spectrum*, 43(4), p. 877–904.

Toth, P. & Vigo, D. eds., 2002. *The Vehicle Routing Problem.* s.l.:s.n.

Toth, P., Vigo, D. & Irnich, S., 2014. *Vehicle Routing Problems, Methods, and Applications.* 2nd ed. Bologna, Italy: SIAM.

Tricia, S. M., Stephen, D. L. & MaryLynn, T. Q., 2012. Case study research. In: D. L. Stephen, T. Q. MaryLynn & J. R. Frances, eds. *Qualitative Research and Intoduction to Metods and Designs.* San Francisco: Jossey-Bass, pp. 243-270.

Unni, S., Nair, V. S. & Gunasekar, S., 2015. *E-retail and online customer satisfaction: Differential impact of delivery time in Indian metros.* Piscataway, N.J., IEEE, p. 1–4.

Annex

About the company

COMPANY PROFILE	
Name	Deloma UG
Logo	deloma
Address	Kleineweferssstr 1
Postal Code	47803
City	Krefeld
Country	Germany
Telephone	+4921514121928
Email	support.shop@deloma.de
Website	https://www.deloma.de/
Year established	2011
In-house working language(s)	German, English
Type of Business	Deloma UG assists local small businesses by providing an E-commerce platform for them to sell their products to the community, as well as additional services such as enterprise resource planning (ERP), logistics, hardware and software support, consulting, advertising, and content creation.

Algorithms

Algorithm 3. **PROCESS INPUT DATA**

Input: List of OrderDeliveryReport

Output: List of Tour

***Initialization of variables**: initialize variables resultList, **t** (Tour)*

1. *for each OrderDeliveryReport // iterate the input list*
2. *   **t** ← check valid values and save properties*
3. *   resultList ← add **t** to the output list.*
4. *End*

Algorithm 4. **FIND ROUTING SOLUTIONS USING JSPRIT**

***Input**: **Input:** t: Tour, boolean*

***Output**: Collection<VehicleRoute>*

***Initialization of variables:** initialize variables **routes**, **problem**: VehicleRoutingProblem*

1. ***problem** ← add vehicle from depot obtained from t.*
2. ***for each** TourStopData of t*
3. *   **problem** ← add created Services // see Algorithm 5*
4. ***routes** ←find the fitted algorithm and best solution for the **problem***

End

Algorithm 5. **CREATE SERVICES FOR DELIVERY LOCATIONS (JSPRIT)**

***Input:**List<TourStopData>tds, boolean*

***Output**: Set<Service>*

1. *Initialization of variables: initialize variables s, la, tw*
2. ***for each** td in tds // iterate the input list*
3. *   **s** ← add capacity / demand*
4. *   **s** ← add location coordinates*
5. *   **s** ← add service time*
6. *   **If a then Apply LA (Algorithm 1) else go step 11.***
7. *   **s** ← add time window **tw***

end

Restrictions for the data collection

If the dataset meets the following conditions, then they are excluded from this study.

the tour day covers more than one day,

- having less than 10 delivery locations,
- missing Location data, Time window data, and Quantity of ordered items are missing.
- orders that are not served (cancelled or in the process) in other words the tours that are postponed.
- orders that are placed earlier than in the year 2021.
- Suppliers or depots are other than specific ones that supply the region that relates to the case study selected shops.

List of other Restrictions

In this paper following constraints are considered:

- Single depot, from which a heterogeneous fleet vehicle with a fixed load capacity and costs starts but must not return.
- Customers are known and have
  - a selected time window during which they must be served.
  - a fixed location where the vehicle must visit.
  - a fixed demand (quantity of item units), historical data that corresponds to the priority of time window flexibility input parameters.
- The delivery must occur in presence of the customer.
- For a single route, same-day deliveries: The corresponding day doesn't overlap with the next or previous (Starts and ends between 00:01 and 23:59).
- In this study, only a single time window with a valid range is used. Customers having multiple time windows for example 10:00 to 12:00 and 15:00 to 20:00 are not considered and can be done in future research.
- Delivery locations are fixed and in a radius of 30km from the depot.
- Costs: fixed.
- Time windows larger than 3 hours are exempted from manipulation in this study.
- New customers are prioritized by default.
- The tour is planned for one single day, not overlapping days.

Results (Graphs and tables)

Case 1 Vs Case 2

Table 5: Case 1 vs Case 2 overall comparison.

Variable	*Case 1*	*Case 2*	*Difference*
R_{po} (%)	63.70	99.30	+ 35.60
S_{dist_sum} (km)	41715.70	40058.00	- 1657.70
S_{dur_sum} (hours)	1084.73	1096.18	+12.05

Case 1 vs Case 3V1

Table 6: Case 1 vs Case 3V1 overall comparison.

Variable	*Case 1*	*Case 3V1*	*Difference*
R_{po} (%)	63.7	65.10	+1.4
S_{dist_sum} (kms)	41715.70	37431.00	-4284.7
S_{dur_sum} (hours)	1084.73	1038.72	-46.01

Case 1 vs Case 3V2

- **Based on sales volumes ($S_{vol\_current}$ and $S_{vol\_hist\_avg}$)**

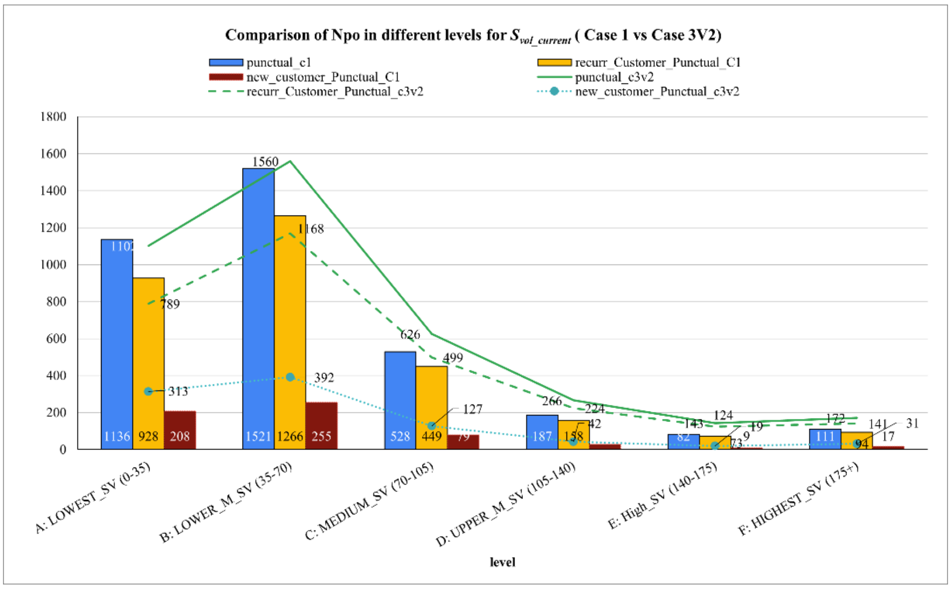

Figure 17: Comparison of Npo based on levels (Svol_current) for Case 1 vs Case 3V2

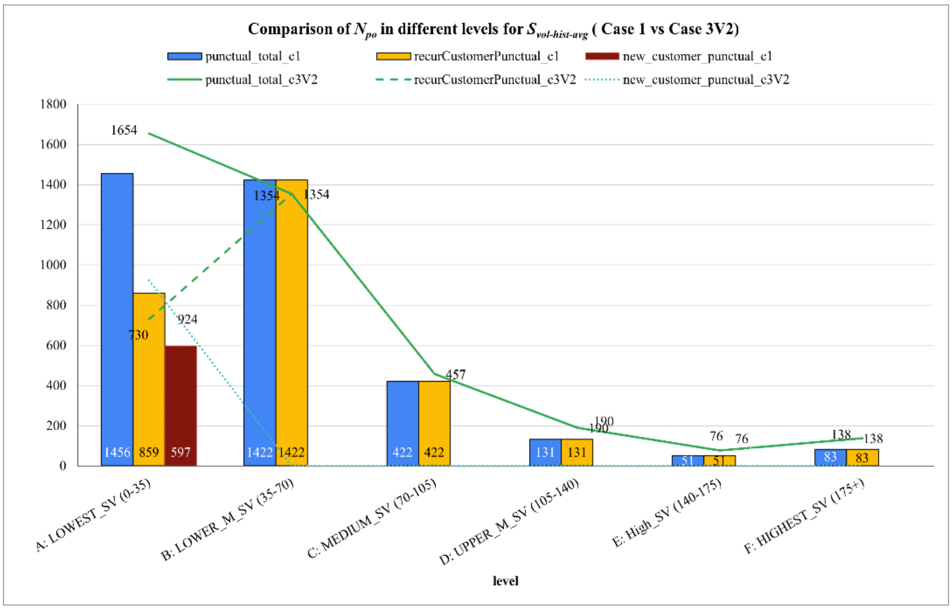

Figure 19:Comparison of Npo based on levels (Svol_hist_avg) for Case 1 vs Case 3V2

- **Based on the number of orders (N_{o_hist})**

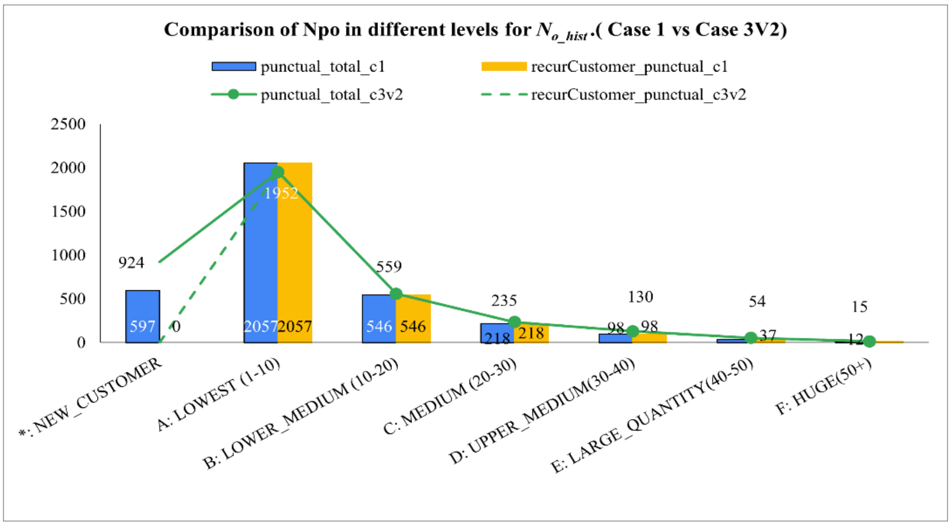

Figure 18: Comparison of Npo based on levels (No_hist) for Case 1 vs Case 3V2

Table 7: Case 1 vs Case 3V2 overall comparison.

Variable	**Case 1**	**Case 3V2**	**Difference**
Rpo (%)	63.7	68.80	+5.1
Sdist_sum (kms)	41715.70	37398.00	-4317.7
Sdur_sum (hours)	1084.73	1043.03	-41.7

Case 1 vs Case 3V3

- **Based on the number of orders (N_{o_hist})**

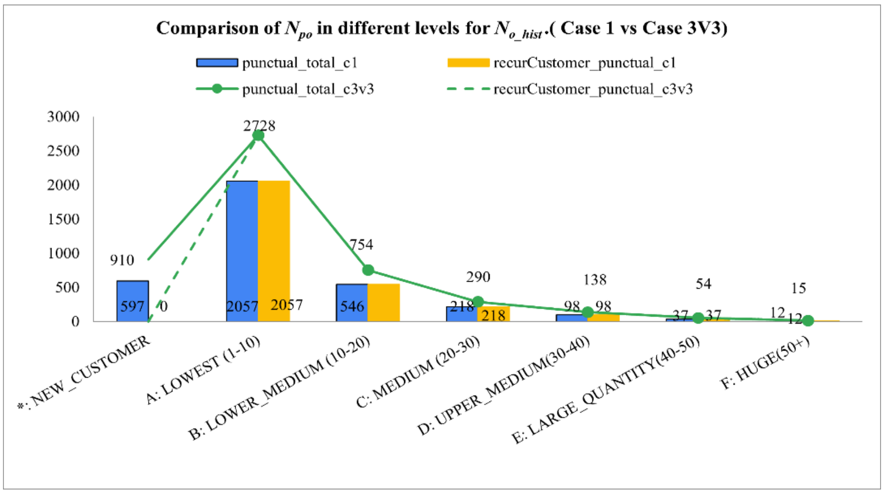

Figure 20:Comparison of Npo in different levels for No_hist (Case 1 vs Case 3V3)

- **Based on sales volume ($S_{vol\_current}$ and $S_{vol\_hist\_avg}$)**

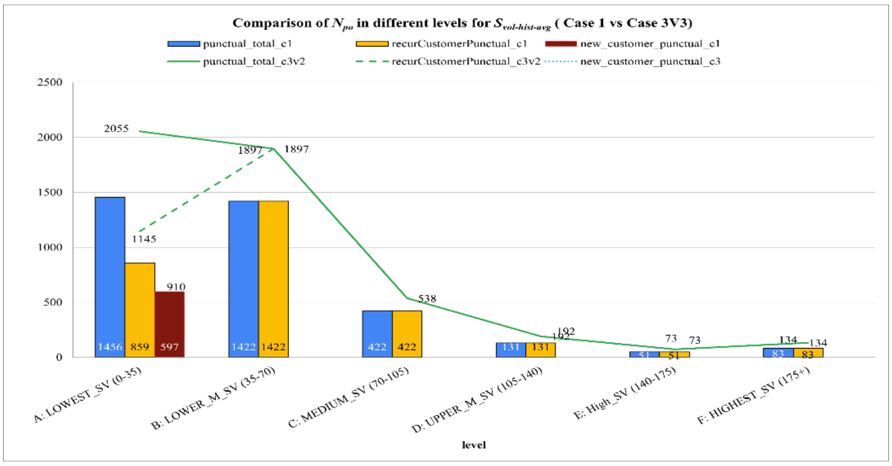

Figure 22:Comparison of Npo in different levels for Svol_current (Case 1 vs Case 3V3).

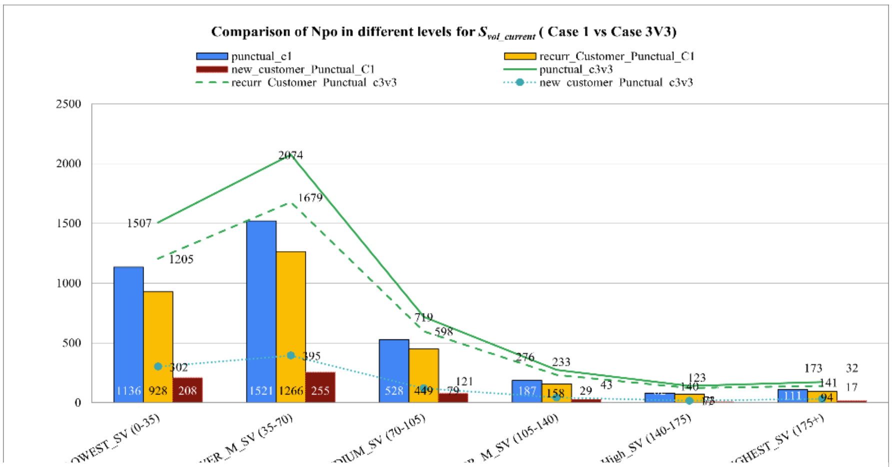

Figure 21: Comparison of Npo in different levels for Svol_hist_avg (Case 1 vs Case 3V3).

- **Based on customer type (C_{type})**

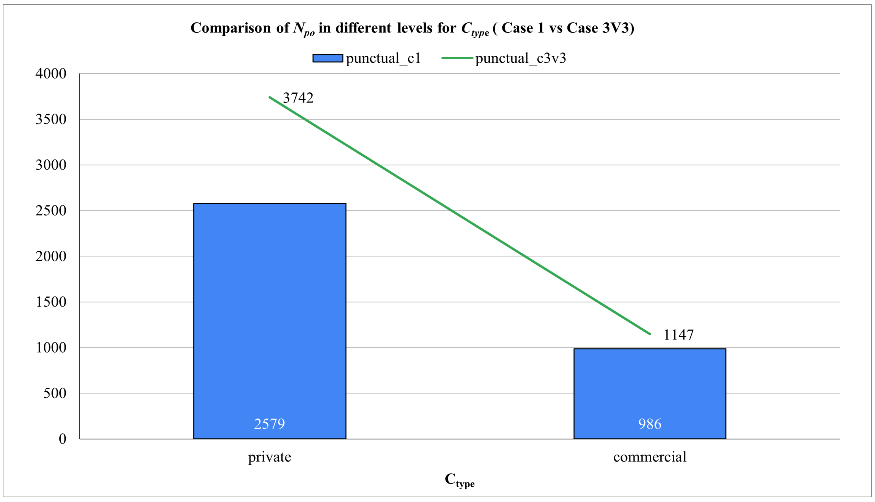

Figure 23:Comparison of Npo in different levels for Ctype (Case 1 vs Case 3V3).

- **Based on E' and F'**

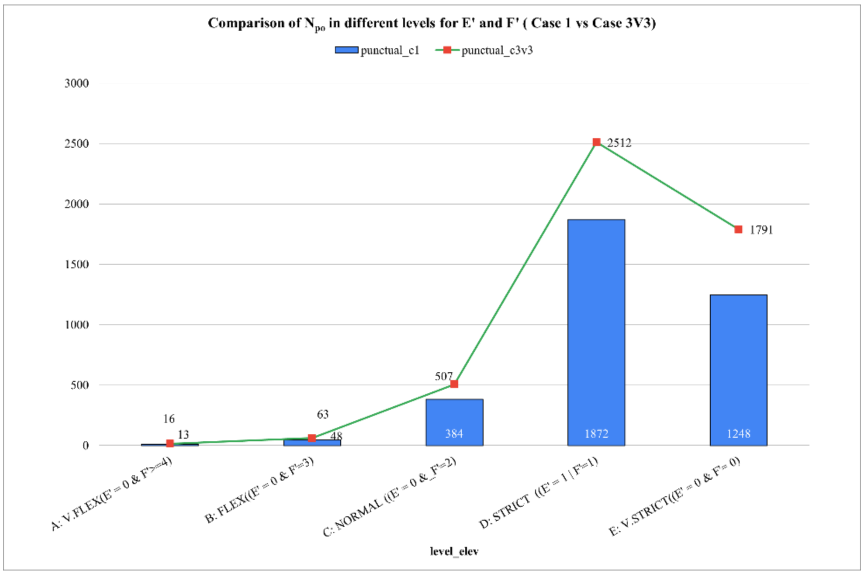

Figure 24: Comparison of Npo in different levels for E' and F' (Case 1 vs Case 3V3).
Here E' = 0 means there is no elevator and F' is the number of floors.

Table 8: Case 1 vs Case 3V3 overall comparison.

Variable	Case 1	Case 3V3	Difference
R_{po} (%)	63.7	87.00	**+23.3**
S_{dist_sum} (km)	41715.70	38957.00	**-2758.7**
S_{dur_sum} (hours)	1084.73	1070.85	**-13.88**

Table 9: Setting weighted factors (coefficient for individual score) for different versions.

f_s	f_p	f_{sv}	f_{OT}	f_{CT}	f_{EF}
V1	1	0	0	0	0
V2	0	1	1	0	0
V3	1	1	1	1	0.5

Overall scenario

For a significant value of 0.05, t-Test results for different independent variables are shown in the following table. Here, $H_0 = 0$ means the Null hypothesis is rejected, else retrains.

Table 10: t-Test: Paired Two Samples for Means results of all variable samples

Variable	Case	Mean (M)	Variance	t (399)	P (T<=t) (Two tail)	P(T<=t) < 0.05	H0
R_{po}	Case 1	8.91	7.76	-	-	-	0
	Case 2	13.90	5.73	-38.75	2.47E-137	TRUE	0
	Case 3V1	9.22	11.02	-1.62	0.10	FALSE	1
	Case 3V2	9.68	7.73	-4.44	1.15E-5	TRUE	0
	Case 3V3	12.22	7.16	-22.50	5.48E-73	TRUE	0
S_{dist}	Case 1	104.29	927.91	-	-	TRUE	0
	Case 2	100.15	1234.66	4.60	5.74E-06	TRUE	0
	Case 3V1	93.58	1178.86	12.02	1.34E-28	TRUE	0
	Case 3V2	93.50	1106.74	12.57	8.92E-31	TRUE	0
	Case 3V3	97.39	1199.58	8.03	1.08E-14	TRUE	0
S_{dur}	Case 1	162.71	1066.49	-	-	TRUE	0
	Case 2	164.43	1491.97	-1.87	0.06	FALSE	1
	Case 3V1	155.80	1472.87	7.29	1.58E-12	TRUE	0
	Case 3V2	156.46	142069	6.82	3.30E-11	TRUE	0
	Case 3V3	160.63	1459.24	2.34	0.02	TRUE	0

Table 11: Mean value differences (in per cent) of variables compared to Case 1

Variable	*Case 2*	*Case 3V1*	*Case3V2*	*Case 3V3*
R_{po_mean} (%)	**+56.0**	+3.5	+8.9	**+37.1**
S_{dist_mean} (%)	-4.0	-10.3	**-10.3**	-6.6
S_{dur_mean} (%)	+1.0	**-4.2**	-3,8	**-1.3**

Appendix

Declaration of Authenticity

I, Azahar Hossain, hereby declare that the work presented herein is my own work completed without the use of any aids other than those listed. Any material from other sources or works done by others has been given due acknowledgement and listed in the reference Section. Sentences or parts of sentences quoted literally are marked as quotations; identification of other references regarding the statement and scope of the work is quoted. The work presented herein has not been published or submitted elsewhere for assessment in the same or a similar form. I will retain a copy of this assignment until after the Board of Examiners has published the results, which I will make available on request.

---	---
Place and Date	Signature: Azahar Hossain